Lecture Notes in Electrical Engineering 936

Pradeep Kumar Singh
Sławomir T. Wierzchoń
Jitender Kumar Chhabra
Sudeep Tanwar *Editors*

Futuristic Trends in Networks and Computing Technologies

Select Proceedings of Fourth International Conference on FTNCT 2021

Springer

Lecture Notes in Electrical Engineering

Volume 936

The book series *Lecture Notes in Electrical Engineering* (LNEE) publishes the latest developments in Electrical Engineering - quickly, informally and in high quality. While original research reported in proceedings and monographs has traditionally formed the core of LNEE, we also encourage authors to submit books devoted to supporting student education and professional training in the various fields and applications areas of electrical engineering. The series cover classical and emerging topics concerning:

- Communication Engineering, Information Theory and Networks
- Electronics Engineering and Microelectronics
- Signal, Image and Speech Processing
- Wireless and Mobile Communication
- Circuits and Systems
- Energy Systems, Power Electronics and Electrical Machines
- Electro-optical Engineering
- Instrumentation Engineering
- Avionics Engineering
- Control Systems
- Internet-of-Things and Cybersecurity
- Biomedical Devices, MEMS and NEMS

For general information about this book series, comments or suggestions, please contact leontina.dicecco@springer.com.

To submit a proposal or request further information, please contact the Publishing Editor in your country:

China

Jasmine Dou, Editor (jasmine.dou@springer.com)

India, Japan, Rest of Asia

Swati Meherishi, Editorial Director (Swati.Meherishi@springer.com)

Southeast Asia, Australia, New Zealand

Ramesh Nath Premnath, Editor (ramesh.premnath@springernature.com)

USA, Canada:

Michael Luby, Senior Editor (michael.luby@springer.com)

All other Countries:

Leontina Di Cecco, Senior Editor (leontina.dicecco@springer.com)

**** This series is indexed by EI Compendex and Scopus databases. ****

Pradeep Kumar Singh · Sławomir T. Wierzchoń ·
Jitender Kumar Chhabra · Sudeep Tanwar
Editors

Futuristic Trends in Networks and Computing Technologies

Select Proceedings of Fourth International Conference on FTNCT 2021

Springer

Editors
Pradeep Kumar Singh
KIET Group of Institutions
Ghaziabad, Uttar Pradesh, India

Jitender Kumar Chhabra
Department of Computer Engineering
NIT Kurukshetra
Kurukshetra, Haryana, India

Sławomir T. Wierzchoń
Institute of Computer Science
Polish Academy of Sciences
Warsaw, Poland

Sudeep Tanwar
Department of Computer Science
and Engineering
Institute of Technology
Nirma University
Ahmedabad, Gujarat, India

ISSN 1876-1100 ISSN 1876-1119 (electronic)
Lecture Notes in Electrical Engineering
ISBN 978-981-19-5036-0 ISBN 978-981-19-5037-7 (eBook)
https://doi.org/10.1007/978-981-19-5037-7

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Organising Committee

Patron

Shri. K. K. Patel, Vice President, Nirma University, India
Dr. Anup Singh, Director General, Nirma University, India

Co-patrons

Dr. R. N. Patel, Director, Institute of Technology, Nirma University, India
Shri. G. Ramachandran Nair, Executive Registrar, Nirma University, India

Honorary Chairs

Dr. Juan Manuel Dodero, University of Cádiz, Spain
Dr. Valeriy Vyatkin, Luleå University of Technology, Sweden
Dr. Sandeep Joshi, Manipal University, Jaipur, India
Dr. Juan José Domínguez Jiménez, University of Cádiz, Spain
Dr. Hernán D. Merlino, University of Buenos Aires, Argentina
Dr. Raúl Saroka, University of Buenos Aires, Argentina
Dr. Roman Mesheryakov, Institute of Control Sciences of Russian Academy of Sciences, Russia
Dr. Miriam Nicado García, University of Habana, Cuba
Dr. Walter Baluja García, University of Informatics Sciences, Cuba
Dr. Jakov Korovin, Southern Federal University, Russia
Dr. Vladimir Kureychik, Southern Federal University, Russia
Dr. Konstantin Rumyantsev, Southern Federal University, Russia
Dr. Michael Karyakin, Southern Federal University, Russia

Dr. Evgeny Abramov, Southern Federal University, Russia
Dr. Ján Labun, Technical University of Kosice, Slovakia
Dr. Pavol Kurdel, Technical University of Kosice, Slovakia
Dr. Jose Francisco Chicano Garcia, University of Malaga, Spain

Principal General Chairs

Dr. Madhuri Bhavsar, CSE, Institute of Technology, Nirma University, India
Dr. Yuriy Zachinyaev, Southern Federal University, Russia
Dr. Konstantin Rumyantsev, Southern Federal University, Russia

Executive General Chairs

Dr. Anton Pljonkin, Southern Federal University, Russia
Dr. Pradeep Kumar Singh, Jaypee University of Information Technology, India

General Chair

Dr. Sudeep Tanwar, CSE, Institute of Technology, Nirma University, India

Co-general Chairs

Dr. Ankit Thakkar, CSE, Institute of Technology, Nirma University, India
Dr. Vijay Ukani, CSE, Institute of Technology, Nirma University, India
Dr. Zunnun Narmawala, CSE, Institute of Technology, Nirma University, India

Organising Chairs

Dr. Aleksey Samoilov, Southern Federal University, Russia
Dr. Abhijit Sen, CS and Information Technology, Kwantlen Polytechnic University, Canada
Dr. Maria Ganzha, University of Technology, Warsaw, Poland
Dr. Marcin Paprzycki, Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland

Publication Chairs

Dr. Rupal Kapadi, CSE, Institute of Technology, Nirma University, India
Prof. Parita Oza, CSE, Institute of Technology, Nirma University, India
Dr. Jitender Kumar Chhabra, Department of Computer Engineering, NIT Kurukshetra, India
Dr. Narottam Chand Kaushal, NIT Hamirpur, India
Dr. Sanjay Sood, Associate Director, C-DAC, Mohali, India

Publicity Chairs

Dr. Ankit Thakkar, CSE, Institute of Technology, Nirma University, India
Dr. Jai Prakash Verma, CSE, Institute of Technology, Nirma University, India
Dr. Ioan-Cosmin Mihai, "Alexandru Ioan Cuza" Police Academy, Romania
Dr. Pelin Angin, Purdue University, USA
Dr. Sudeep Tanwar, Nirma University, India

Program Chairs

Dr. Sharda Valiveti, CSE, Institute of Technology, Nirma University, India
Prof. Pronaya Bhattacharya, CSE, Institute of Technology, Nirma University, India

Organising Secretary

Shri. B. J. Patel, Institute of Technology, Nirma University, India
Dr. Swati Jain, CSE, Institute of Technology, Nirma University, India
Dr. Smita Agrawal, CSE, Institute of Technology, Nirma University, India
Dr. Yugal Kumar, Jaypee University of Information Technology, India
Dr. Sudhanshu Tyagi, Thapar Institute of Engineering and Technology, Patiala, India

Track Chairs

Dr. Gaurang Rawal, CSE, Institute of Technology, Nirma University, India
Dr. Priyank Thakkar, CSE, Institute of Technology, Nirma University, India

Sponsorship Committee

Prof. Tejal Upadhyay, CSE, Institute of Technology, Nirma University, India
Dr. Saurin Pareikh, CSE, Institute of Technology, Nirma University, India
Prof. Vishal U. Parikh, CSE, Institute of Technology, Nirma University, India
Prof. Ajay M. Patel, CSE, Institute of Technology, Nirma University, India
Prof. Pooja P. Shah, CSE, Institute of Technology, Nirma University, India

Website Committee

Dr. Rajan Datt, CSE, Institute of Technology, Nirma University, India
Prof. Anuja Nair, CSE, Institute of Technology, Nirma University, India

Registration and Logistics Committee

Prof. Vipul Chudasama, CSE, Institute of Technology, Nirma University, India
Prof. Pimal Khanpara, CSE, Institute of Technology, Nirma University, India

Local Organising Committee

Prof. Sonia Mittal, CSE, Institute of Technology, Nirma University, India
Prof. Monika Shah, CSE, Institute of Technology, Nirma University, India
Prof. Deepika Shukla, CSE, Institute of Technology, Nirma University, India
Prof. Malaram Kumhar, CSE, Institute of Technology, Nirma University, India
Prof. Devendra Vashi, CSE, Institute of Technology, Nirma University, India
Dr. Jigna A. Patel, CSE, Institute of Technology, Nirma University, India
Prof. Vipul Chudasama, CSE, Institute of Technology, Nirma University, India
Dr. Usha Patel, CSE, Institute of Technology, Nirma University, India
Prof. Jitali Patel, CSE, Institute of Technology, Nirma University, India
Prof. Kruti Lavingia, CSE, Institute of Technology, Nirma University, India
Prof. Priti Kathiria, CSE, Institute of Technology, Nirma University, India
Prof. Tarjni Vyas, CSE, Institute of Technology, Nirma University, India
Prof. Shivani Desai, CSE, Institute of Technology, Nirma University, India
Prof. Vivek K. Prasad, CSE, Institute of Technology, Nirma University, India

Academic Collaborators

Nirma University, Ahmedabad, India

SETIT, Tunisia

University of Economy, WSG, Bydgoszcz, Poland

Sothern Federal University, Russia

Other Supporters for the FTNCT-2021: Conference alerts as a technical promoters, IAC Education, India, Easy Chair, and many more.

Preface

The 4th International Conference on Futuristic Trends in Networks and Computing Technologies (FTNCT-2021) was hosted at the venue of the Department of Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad, India. Nirma University is duly recognised by the University Grants Commission (UGC) under Section 2 (f) of the UGC Act. The university is accredited by the National Assessment and Accreditation Council (NAAC). The goal of the FTNCT conference is to explore the challenges, issues, and opportunities in different areas of computing and networking technologies year after year. Last few proceedings have already addressed many latest challenges. The organising team is happy to share that FTNCT-2018 proceedings has 16,000 downloads and 76 citations, FTNCT-2019 proceedings has 25,000 downloads and 43 citations, and FTNCT-2020 proceedings has 14,000 downloads and 13 citations as per Springer webpage information.

The organising team proudly announce that the conference has established a good repute and confidence among the readers in the domain of computing and networks, and we hope in coming years, the conference will emerge as one of the top ranked conferences. The current version of the conference received research papers in four technical tracks, namely network and computing technologies, wireless networks and Internet of Things (IoT), futuristic computing technologies and communication technologies, and security and privacy. The conference is planned with the vision to invite the researchers across the globe to share their researches, methodologies, algorithms, architectures, and research applications during this ongoing annual event.

The 4th International Conference on Futuristic Trends in Networks and Computing Technologies (FTNCT-2021) was hosted by the Department of Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad, Gujarat, India, from 10 to 11 December 2021. During this conference, there were several academic partners and universities, such as SETIT Sfax University, Tunisia, University of Economy WSG, Bydgoszcz, Poland, and many more universities and institutes also contributed directly or indirectly. We express our sincere gratitude to the valuable authors for their contribution and towards the technical program committee members of FTNCT-2021 for their immense support and motivation towards making the fourth version of FTNCT conference as a grand success.

The welcome address was delivered by Dr. P. V. Patel from the Institute of Technology, Nirma University, followed by Principal General Chair, Dr. Madhuri Bhavsar, Head, CSE, Institute of Technology, Nirma University. The inaugural address was given by Chief Guest, Shri. Nilesh Desai, Director, Space Applications Centre, Ahmedabad, in continuation by Guest of Honour, Dr. Rajeev Sharma, Scientist-F, Frontier and Futuristic Technologies (FFT) Division, Department of Science and Technology (DST), Government of India. The presidential address was delivered by Patron, Dr. Anup Singh, Director General, Nirma University, and finally, the vote of thanks of inaugural and valedictory sessions were given by the FTNCT-2021, General Chair, Dr. Sudeep Tanwar, Department of CSE, Institute of Technology, Nirma University, and by FTNCT-2021, Co-general Chair, Dr. Vijay Ukani, Department of CSE, Institute of Technology, Nirma University, respectively.

We are also grateful to our keynote speakers for the delivery of wonderful talks followed by discussions with authors. The first keynote was delivered by Prof. Yassine Maleh, National School of Applied Sciences at Sultan Moulay Slimane University, Morocco, on “Machine Intelligence and Data Analytics for Cybersecurity in the Industry 4.0 Era”; second keynote was given by Prof. Fausto Pedro García Márquez, Full Professor, at Castilla-La Mancha University, Spain, on “Artificial Intelligence and Green Energy”; and third keynote was delivered by Prof. Paolo Rocca, Associate Professor, Department of Information Engineering and Computer Science, University of Trento, Italy, on “Future on mobile communications beyond 5G”. In addition to this, one workshop-cum-technical talk was addressed by Ms. Kamiya Khatter, Editor, Springer Nature India, New Delhi, India, on topic “Scientific Writing for Journal Publication”. Finally, Dr. Pietro Bartocci, University of Perugia, Perugia, Italy, Editor-in-Chief, Energy Nexus, addressed the audience on “Sustainable Green Energy”.

The conference sessions were conducted in the presence of various session chairs, namely Dr. Neerendra Kumar, Central University, Jammu, India; Dr. Yugal Kumar, Jaypee University of Information Technology, Waknaghat, India; Dr. Manish Khare, DAIICT, Gandhinagar, India; Dr. Charu Bhardwaj, TT consultants, Chandigarh, India; Dr. Rohit Tanwar, University of Petroleum and Energy Studies, Dehradun, India; Dr. Bhaskar Mondal, National Institute of Technology Patna, India; Dr. Noor Zaman, Taylor’s University, Malaysia; Dr. Sudhanshu Tyagi, Thapar University, Patiala, India; Dr. Samayveer Singh, National Institute of Technology Jalandhar, India; Dr. Amit Sharma, Chitkara University, Chandigarh, India; Dr. Zunnun Narmawala, Institute of Technology, Nirma University, Ahmedabad, India; Dr. Rachna Jain, Bhagwan Parshuram Institute of Technology, NCR, India; Dr. Nagesh Kumar, Chitkara University, Chandigarh, India; Dr. Vivek Sehgal, Jaypee University of Information Technology, Waknaghat, India; Dr. Rakesh Vanzara, U. V. Patel College of Engineering, Mehsana, India; Dr. Mayank Pandey, Motilal Nehru National Institute of Technology, Allahabad, India; Dr. Sachin Kumar, Amity University, Lucknow, India; Dr. Karan Singh, Jawaharlal Nehru University, New Delhi, India; Dr. Shaweta Khanna, ITS, Greater Noida, India; Dr. Bramah Hazela, Amity University, Lucknow, India; Dr. Hitesh Chhinkaniwala, Adani Institute of Infrastructure Engineering, Ahmedabad, India; Dr. Amit Kumar, Jaypee University of Information

Technology, Waknaghat, India; Dr. Preeti Nagrath, Bharti Vidyapeeth College of Engineering, New Delhi, India; Dr. Manish Chaturvedi, Institute of Infrastructure Technology Research and Management, Ahmedabad, India; Dr. Yashwant Singh, Central University, Jammu, India; Dr. Rakesh Saini, DIT University, Dehradun, India; Dr. Anand Nayyar, Graduate School, Duy Tan University, Da Nang, Vietnam; Dr. Ashwani Kumar, Sreyas Institute of Engineering and Technology, Hyderabad, India; Dr. Anton Pavlovich Pljonkin, Southern Federal University, Russia; Dr. Sanjay Garg, Indrashil University, Kadi, India, along with Co-general Chair Dr. Vijay Ukani, Institute of Technology, Nirma University, and many anonymous session chairs and session rapporteurs. The organising team would like to mention special thanks to the patrons: Shri. K. K. Patel, Vice President, Nirma University, India, and Dr. Anup Singh, Director General, Nirma University, India, along with Co-patrons: Dr. R. N. Patel, Director, Institute of Technology, Nirma University, India, Shri. G. Ramachandran Nair, Executive Registrar, Nirma University, India, for their support, guidance, and many administrative approval from time to time.

We express our sincere gratitude to our publication partner, Springer, LNEE Series, for believing in us. We are thankful to Ms. Kamiya Khatter, Executive Editor, Springer, for extending her help from time to time for the preparation of this proceedings.

Ghaziabad, India	Pradeep Kumar Singh
Warsaw, Poland	Sławomir T. Wierzchoń
Haryana, India	Jitender Kumar Chhabra
Ahmedabad, India	Sudeep Tanwar
December 2021	

Contents

Network and Computing Technologies

Wireless Networks and Internet of Things (IoT)

Futuristic Computing Technologies

Communication Technologies, Security and Privacy

About the Editors

Pradeep Kumar Singh is a Professor and Head in the Department of Computer Science at KIET Group of Institutions, Delhi-NCR Campus, India. He is recently appointed as a Section Editor, *Discover IoT*, Springer Journal. He has published over 120 research papers. Dr. Singh has received three sponsored research project grants worth Rs. 25 Lakhs. He has edited a total of 16 books and also edited several special issues for SCI and SCIE Journals published by reputed international publishers.

Sławomir T. Wierzchoń received M.Sc. and Ph.D. degrees in Computer Science from the Technical University of Warsaw, Poland. He holds a Habilitation (D.Sc.) in Uncertainty Management from the Polish Academy of Sciences. In 2003 he received the title of Professor from the President of Poland. Currently, he is a full professor at the Institute of Computer Science of the Polish Academy of Sciences. His research interests include computational intelligence, uncertainty management, information retrieval, machine learning, and data mining. He is an author/co-author of over 100 peer-reviewed papers in international journals and international conferences. He published, as author/co-author, 11 monographs from the field of Machine Learning.

Jitender Kumar Chhabra is a Professor in the Computer Engineering Department at the National Institute of Technology, Kurukshetra, India. He has published 120 papers in reputed International and National Journals and conferences including over 40 publications. His research interest areas are software metrics, data mining, soft computing, machine learning, algorithms and related areas. He is a reviewer for most reputed journals such as *IEEE Transactions*, *ACM Transactions*, etc.

Sudeep Tanwar (Senior Member, IEEE) is working as a full professor at the Nirma University, India. He is also a Visiting Professor with Jan Wyzykowski University, Poland, and the University of Pitesti, Romania. He received B.Tech in 2002 from Kurukshetra University, India, M.Tech (Honor's) in 2009 from Guru Gobind Singh Indraprastha University, Delhi, India and Ph.D. in 2016 with specialization in Wireless Sensor Network. He has authored 04 books and edited 20 books, more than 300 technical articles, including top cited journals and conferences, such as IEEE TNSE, IEEE TVT, IEEE TII, IEEE TGCN, IEEE TCSC, IEEE IoTJ, IEEE NETWORKS, IEEE WCM, ICC, IWCMC, GLOBECOM, CITS, and INFOCOM. He initiated the research field of blockchain technology adoption in various verticals in 2017. His H-index as per Google Scholar and Scopus is 56 and 45, respectively. His research interests include blockchain technology, wireless sensor networks, fog computing, smart grid, and the IoT. He is a member of the Technical Committee on Tactile Internet of IEEE Communication Society. Recently, He has been awarded a cash prize of Rs, 50,000 for publishing papers with 5+ Impact factor and publication of books with Springer, IET & CRC under the scheme of "Faculty Awards and Incentives" of Nirma University for the year 2019–2020. He has been awarded the Best Research Paper Awards from IEEE IWCMC-2021, IEEE ICCCA-2021, IEEE GLOBECOM 2018, IEEE ICC 2019, and Springer ICRIC-2019. He has won Dr KW Wong Annual Best Paper Prize for 2021 sponsored by Elsevier (publishers of JISA). He has served many international conferences as a member of the Organizing Committee, such as the Publication Chair for FTNCT-2020, ICCIC 2020, and WiMob2019, and a General Chair for IC4S 2019, 2020, ICCSDF 2020, FTNCT 2021. He is also serving the editorial boards of *COMCOM-Elsevier, IJCS-Wiley, Cyber Security and Applications-Elsevier, Frontiers of blockchain*, and *SPY, Wiley*. He is also leading the ST Research Laboratory, where group members are working on the latest cutting-edge technologies.

Network and Computing Technologies

Classification of Hate Tweets Using Hybrid Deep Belief Network Algorithm

Pramod Sunagar, Anita Kanavalli, Sushmitha S. Nayak, Shriya Raj Mahan, Saurabh Prasad, and Shiv Prasad

Abstract Social media platforms have presented a way to express the users' opinions on various topics and connect to friends and share messages, photos, and videos. But there has been an increase in abusive, racial, and hateful messages. As a result, hate tweets have become a significant issue in social media. Detecting hate tweets from Twitter posts with little contextual detail poses several practical problems. Furthermore, the variety of user-generated information and the existence of different hate speech make determining the degree and purpose of the post extremely difficult. A deep belief network with softmax regression is implemented in this work utilizing various embedding techniques for detecting hate speeches in social media. A deep belief network is chosen for resolving the sparse high-dimensional matrix estimation hitch of the text data. Softmax regression is executed to classify the text data in the provided learned feature space, succeeding the feature extraction procedure using hybrid DBN. Experiments are performed on the publicly accessible dataset and evaluate the effectiveness of the deep learning model by considering various metrics.

Keywords Deep belief network · Hate tweets · Limited-memory BFGS · Restricted Boltzmann machine · Softmax regression · Text classification

1 Introduction

Due to the unprecedented growth of social media, individuals may now exchange and distribute ideas at an unprecedented rate. Although interactions on social networking sites may increase an individual's sense of connectivity with real and virtual groups, the same platforms are increasingly being used to spread poisonous content such as

P. Sunagar (✉) · A. Kanavalli · S. S. Nayak · S. R. Mahan · S. Prasad · S. Prasad
M S Ramaiah Institute of Technology (Affiliated to VTU), Bangalore, India
e-mail: pramods@msrit.edu

A. Kanavalli
e-mail: anithak@msrit.edu

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_1

hate speeches based on race, religion, gender, or sexual orientation. Hate speech on social media has not only fostered strife among people or communities online but has also manifested in violence. As a result, detecting and suppressing hateful speech on online social media platforms is a critical issue. Many major social networking sites have made significant efforts to detect and prevent hate speech posts. Researchers are developing new methods for detecting, categorizing, and removing hateful messages [1, 2]. One of the most significant challenges in the natural language processing disciplines is the classification of hate speech. There is a need for the detection of the spread of hatred via social media platforms. Hence, the category of offensive messages is necessary. Hate posts are words that may be in tweets, social media posts, audio, and videos. These posts are precluded since they lead to acts that trigger savagery and rebel demeanor toward other people or groups. Social media users spread hate messages through their posts on Facebook, Twitter, Instagram, etc. These posts will be used to target individuals, sportspersons, artists, religions, and countries. Hence, detecting and classifying social media posts as hate messages or not is the latest work that needs more breakthroughs. In this work, the tweets are classified as hate tweets or average tweets by incorporating multiple layers softmax regression, a multi-class classifier, along with the deep belief network algorithm that utilizes the restricted Boltzmann machines (RBM) or autoencoders. Hence, deep learning models are used primarily in the research areas of sentiment analysis for higher accuracy as they learn high-level features from the dataset over many epochs. The results obtained were fair enough in hate speech classification.

2 Related Works

Koo investigated fostering a few directed models dependent on DBN to develop this two-stage methodology [3] further. Grouping exactness can be improved by regularizing the model boundaries with the qualities prepared for solo and directed reason. Upgrades to momentum BL can be made, for example, applying distinctive arrangement search calculations, regulated learning regularization procedures, or diverse statement methodologies. Zheng utilized the RCNN with BI-LSTM with Word2Vec [4]. The model consolidates the bidirectional long momentary memory and convolutional neural organization with the consideration system and word2vec to accomplish the fine-grained text grouping task. The proposed Bi-LSTM performed better. An epic crossover text arrangement model dependent on the profound conviction organization and softmax relapse [5] was created by Mingyang Jiang. After the element extraction with DBN, softmax relapse is utilized to arrange the content in the learned element space. In pre-preparing strategies, the profound conviction organization and softmax relapse are first prepared individually. Although DBN supplements the learning usefulness to address the considerable measurement and meager network issue and softmax relapse is utilized to characterize the writings, they will fill in as a steady total model with adjusting and accomplishing the end-product.

Additionally, the proposed technique utilizes the L-BFGS calculation to advance the loads, one of the semi Newton strategies and utilizations, the backward Hessian network assessment. Abdel-Zaher implemented computer-aided diagnosis, which will aid in the early detection of breast cancer using a deep belief network [6]. The proposed model has shown an accuracy of 99.68%, which asserts promising results. Zheng had implemented a deep belief network to categorize the positive and negative emotions from EEG data [7]. A hybrid DBN with a hidden Markov model is implemented in this work, outperforming other models considered.

Models utilizing term recurrence converse archive recurrence and word inserting have been applied to a progression of datasets. At long last, a near report has been directed on the trial results acquired for the various models and info highlights. Likewise, the examinations uncovered that CNN beats different models, introducing a decent harmony among precision and CPU runtime [8]. The NLP procedures referenced in this article eliminate characters, lowercase letters, and roots to diminish word expressions and split the content into tokens. As neural organization approaches beat existing text arrangement issues, a profound learning model called convolutional neural network was presented. Taking a gander at each class independently, it ought to be noticed that numerous scornful tweets have been misclassified. Ideas were made to utilize the document's techniques and progress the arrangement assignments [9]. The pre-preparing strategies employed are case collapsing, stop word expulsion, and radicalism. After the word vector extraction from the extraction of worldwide vector highlights for all datasets, all sentences being shaped utilize the deep belief network, which would then characterize the sentences into two orders of disdain discourse or not disdain discourse [10]. The applications of deep belief networks are found in the biomedical domain [11] and denoising autoencoders [12].

In this paper, the first choices are delivered with TF IDF-ICF, a book grouping model upheld on DBN is constructed. A substitution AI algorithmic standard supported on DBNs is gotten ready for text order. DBN displaying incorporates RBM hypothesis, RBM energy model, DBN instructing, and network adjustment. The outcomes show that the exhibition of the DBN grouping algorithmic standard is significantly higher than that of the SVM algorithmic guideline. For the higher collection of the content, DBN ends up being the easiest [13]. In this paper, the DBN set of rules inside the idea of profound dominating is utilized to separate the top to bottom highlights of the imaging ghostly photo measurements [14]. The novel records are planned to work region with solo acquiring information on techniques through the RBM. Results show that the in-force highlights separated through the significant thought network calculation have better vigor and distinguishableness. Results show that the top to bottom highlights removed through the powerful conviction local area set of rules have higher strength and detachability. DBN is applied to various sorts of data. Continuous data, realities parallelism, and multimodal realities are a portion of the bundles of DBN. It is far a better model of profound acquiring information. Profound becoming acquainted with calculations extricate immense outline portrayals of the crude data through the utilization of various leveled staggered becoming more acquainted with strategy. The removed descriptions might be considered an actual comprehension inventory for significant insights investigation duties, comprising measurements labeling and notoriety records recovery and regular language preparation [15]. Zhang executed profound fake neural organizations with the plan to discover unique component portrayals from input records through its various stacked layers to incite the grouping of the example disdain discourse [16].

CNN + LSTM. It is eye-catching to see that the projected baselines region unit is excellent, eminently the element assigned SVM models and the gauge neural organization models that get on top of the previously announced figures on five datasets thought about against cutting edge. Aftereffects of different CNN + LSTM models also show that the model learns higher with the pre-prepared word implanting, any place the most straightforward F1 is accomplished with emb-ggl1 on five datasets and with emb-ggl2 on two datasets. Disdain discourse via online media has been expanding lately. Indeed, even with decent endeavors from the requirement offices, a significant answer exists in the information handling method. The information sources tend to take changed structures to highlight cryptography just as the outcomes from old-style systems; nonetheless, the most differentiation is that the info alternatives are not utilized straightforwardly for discourse grouping. The work referenced during this paper contributes tons to the goal to this disadvantage.

During hate discourse identification, the absolute first undertaking is to get the component implanting that is progressively made from word inserting and character-n-grams. There are unit two types of inserting—nonstop baggage of terms (CBOW) and skip-gram models. A pooling layer in each organization changes over each tweet into many length vectors, catching all information required from a tweet. In the CNN model, include implanting utilized were arbitrary word vectors while instructing the organization. This pattern model accomplished exactness, review, and F-score upsides of 86.68, 67.26, and 75.63%, severally, denoting a goliath improvement in accuracy contrasted with the LogReg model, notwithstanding to the detriment of lower review. Examinations were made for Twitter disdain discourse recognition upheld profound learning, convolutional neural organization models. The list of capabilities is down-sized inside the organizations by a top pooling layer.

In contrast, a softmax layer is utilized to appoint the tweets as their most plausible name class [17]. The principle point of the work is to characterize the tweets into three classifications: scornful, hostile, and clean. At first, the data gathered is pre-processed, and when this strategy fundamentally works, any place that includes extraction happens. They legitimize that the model is made once by considering three recognized AI calculations for text characterization: strategic regression, Naive Bayes, and support vector machines. We get promising outcomes supporting the TFIDF approach on the pack of words. Models prepared once extricating N-gram choices from the content offer higher results. The aftereffects of the closer examination of strategic regression, Naive Bayes, and support vector machines, and every one of the three calculations perform extensively for the L2 normalization of TFIDF; however, execution of SVM is poor contrasted with the others. Examination of the least difficult exactness for Naive Bayes and calculated regression, strategic regression performs higher. An answer was intended to discover the disdain discourse and hostile language through AI abuse n-gram highlight-weighted with TFIDF esteems [18]. The outcomes are frequently additionally improved by expanding the review for the aggressive class and exactitude for the disdainful classification, and upgrades are regularly done by consolidating semantic alternatives.

Two principal assignments were performed while achieving the obligation. To begin with, it is approached to have a twofold order of the tweets, which is as either sick natured. The exhibition of the framework is estimated to help the exactness. Then, it is approached to characterize the misogynous tweets per misanthropic conduct and the message's objective. The investigation metric is a full-scale F1 score for this assignment. We tend to get the most specific outcome for English assignment an in run one (0.704 precision), during which we tend to utilize an arrangement Regression classifier and have the first Rank. During this paper, we tend to give our way to deal with sight-sick-natured tweets on Twitter. We tend to create sentence installing, TF-IDF vectors, and BOW vectors for each tweet, so link them. These vectors are then utilized as alternatives for models like CatBoost and arrangement regression [19]. This model involved the most elevated three situations for English Subtask, and our most prominent model for English Subtask B came at the fifth position (third-best group). The paper refers to teaching models in multilingual settings, and we need multilingual word/sentence implanting. For sentences, laser installing has been utilized, and for words, MUSE embedding has been used. The styles made were CNN-GRU, bidirectional encoder representations from transformers (BERT), and multilingual BERT (MBERT). The outcomes of the monolingual situation, as expected, show that with expanding preparing data, the classifier, generally speaking, execution increments also.

Notwithstanding, the general presentation appears to contrast depending on the language and the model. Laser + LR plays the best in low-guide settings. On the other hand, laser + LR performs extraordinarily in low-helpful asset settings. Notwithstanding how laser + LR is doing appropriately in low-helpful asset settings, if enough realities are to be had, we analyze that BERT fundamentally based designs: Translation + BERT and MBERT are improving. In this paper, the principal enormous scope investigation of multilingual disdain discourse is finished. Profound acquiring information on models is utilized to expand classifiers for multilingual disdain discourse class. Laser + LR are more successful for low valuable assets, while BERT styles are all the more remarkable [20]. To obtain a weak prediction model, machine learning techniques like logistic regression, random forest techniques, and multinomial Naive Bayes are used in this article [21] to extract the comments' text features and frequency features. To achieve the concluding forecast model, gradient boosting is applied to this model. A neural network with bidirectional long short-term memory is also employed in this work. These models are implemented to assist in filtering out abusive comments from social media.

3 Proposed Methodology

The deep belief network is presented to decipher the sparse high-dimensional matrix computation problem used in the many NLP-related areas. The DBN model first performs the feature extraction. Then, the softmax regression is applied to categorize the data in the learned feature space. The DBN and softmax regression are first trained during pre-training techniques. In the fine-tuning steps, the models are converted into an understandable model and optimize the system parameters with Broyden–Fletcher–Goldfarb–Shanno algorithm. Natural language processing techniques can be used to ensure safety protocols on social media. Tweets can be analyzed; hate words can be detected and classified by pipelining NLP functions with classification algorithms. Exploratory data analysis and feature engineering can be used for tracking defaulters.

Restricted Boltzmann machines (RBMs) can be deliberated as a binary type of factor analysis. An RBM model is comprised of visible (v) and hidden (h) layers, with links between layers, as shown in Fig. 1. The correlations between higher-order data are captured by the hidden units which are trained for this purpose. The generative weights are attained using an unsupervised layer-by-layer greedy technique. The RBN preparation method initiates by awarding a vector v to the visible units that transfer values to the hidden layers. This training is known as Gibbs sampling. On the contrary track, the evident unit feedbacks are stochastically initiated to restructure the initial input. At last, these new visible neuron activations are sent, so single-step reconstruction hidden unit activations, h, can be accomplished.

A DBN may be a kind of deep learning network recreated by stacking RBMs, as displayed in Fig. 2. Each RBM module is trained one at a time in an unsupervised way and employs a contrastive divergence method. The training method is performed in a greedy layer-by-layer mode with weight refinement to abstract hierarchical features resulting from the raw input data. The training procedure is supported layer-by-layer while tuning the weight factors using contrastive convergence (CD) to create a balanced assessment of the learning probability. Other than the conditional likelihood

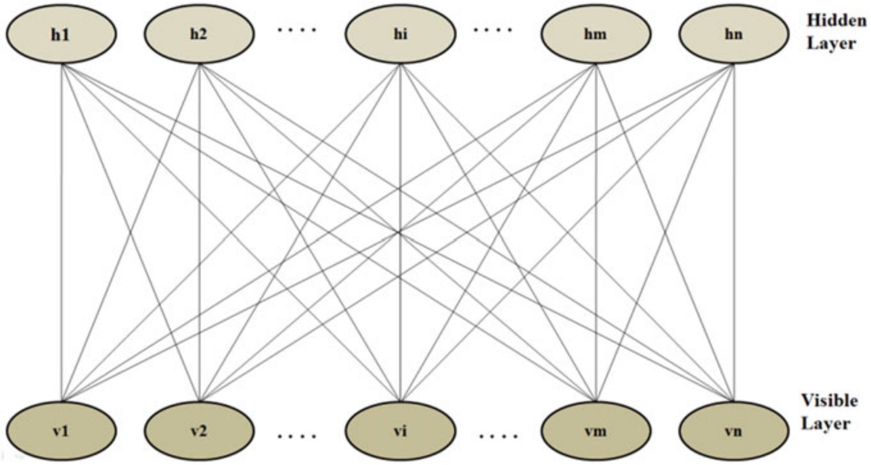


Fig. 1 Restricted Boltzmann machine (RBM)

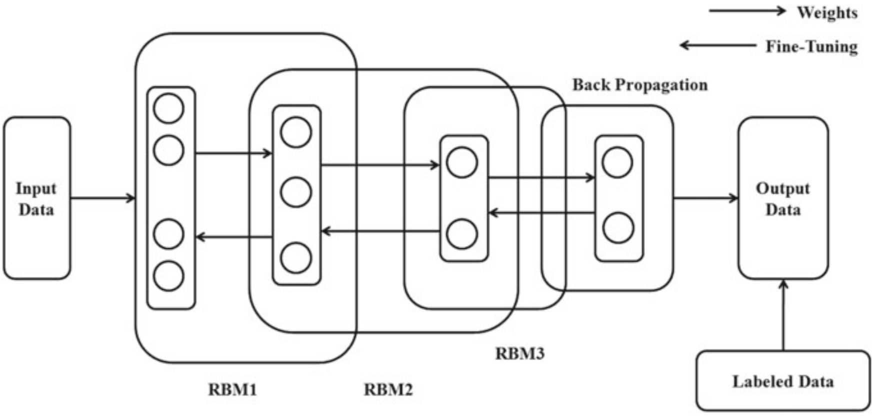


Fig. 2 General architecture of deep belief network

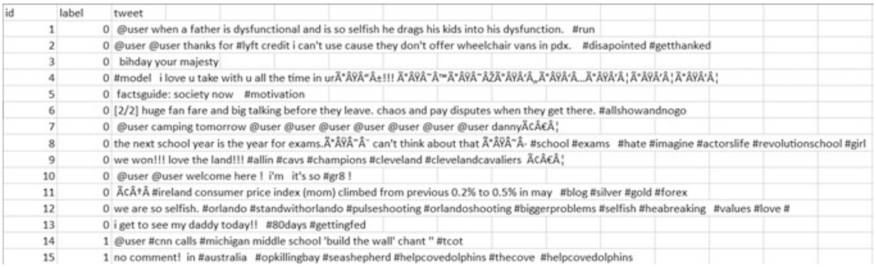
dissemination of the input tests is decided to memorize the unique highlights that are vigorous and perpetual to change, commotion, etc. The output of each step is used as input of the subsequent RBM stage. Later, the entire network is trained with supervised learning to increase accuracy in classification tasks.

4 Results and Discussions

4.1 Dataset

The Twitter sentiment analysis dataset train_E6oV3lV was used for this work, as shown in Fig. 3. The dataset has about 32 k labeled entries. The class label identifies tweets containing hate speech by one and those not containing any hate speech by 0.

The deep belief network was coded by using the Numpy module. A data folder was created to involve the weights of the two RBM layers and softmax. The model



id	label	tweet
1	0	@user when a father is dysfunctional and is so selfish he drags his kids into his dysfunction. #run
2	0	@user @user thanks for #lyft credit i can't use cause they don't offer wheelchair vans in pdx. #disapointed #getthanked
3	0	bihday your majesty
4	0	#model i love u take with u all the time in urÃ°ÂŸÂ"Â±!!! Ã°ÂŸÂ˜Â™Ã°ÂŸÂ˜ÂŽÃ°ÂŸÂ'Â„Ã°ÂŸÂ'Â…Ã°ÂŸÂ'Â¦Ã°ÂŸÂ'Â¦Ã°ÂŸÂ'Â¦
5	0	factsguide: society now #motivation
6	0	[2/2] huge fan fare and big talking before they leave. chaos and pay disputes when they get there. #allshowandnogo
7	0	@user camping tomorrow @user @user @user @user @user @user @user dannyÃ¢Â€Â¦
8	0	the next school year is the year for exams.Ã°ÂŸÂ˜Â¯ can't think about that Ã°ÂŸÂ˜Â #school #exams #hate #imagine #actorslife #revolutionschool #girl
9	0	we won!!! love the land!!! #allin #cavs #champions #cleveland #clevelandcavaliers Ã¢Â€Â¦
10	0	@user @user welcome here ! i'm it's so #gr8 !
11	0	Ã¢Â†Â #ireland consumer price index (mom) climbed from previous 0.2% to 0.5% in may #blog #silver #gold #forex
12	0	we are so selfish. #orlando #standwithorlando #pulseshooting #orlandoshooting #biggerproblems #selfish #heabreaking #values #love #
13	0	i get to see my daddy today!! #80days #gettingfed
14	1	@user #cnn calls #michigan middle school 'build the wall' chant '' #tcot
15	1	no comment! in #australia #opkillingbay #seashepherd #helpcovedolphins #thecove #helpcovedolphins

Fig. 3 Twitter dataset

was tested using 30% of the data from the chosen dataset. The model comprises DBN, which shows an accuracy of <50% when implemented without softmax regression. When softmax regression was implemented along with DBN, the accuracy rate was about 61%. When two softmax stages were employed with the DBN, the accuracy increased to 84%. The accuracy of 94.66% was achieved after the fine-tuning stage. The L-BFGS algorithm and gradient descent algorithm were used to achieve this. A comparison of the DBN models with variations is shown in Fig. 4. The classification metrics such as accuracy (94.66%), precision (99.78%), recall (95%), F1-score (97.2%), sensitivity (94.75%), and specificity (90.57%) were used to evaluate the model. The same is shown in Fig. 5.

The proposed system works well for various sizes of datasets due to integrating a neural network with multi-class regression. The regression had more importance, while the model was built for smaller datasets. DBN came across a better learning rate and accuracy when the same model was used over large or real-time data, as shown in Fig. 4. This characteristic of our model makes it more flexible and precise over the other neural network models in the comparison. We further looked into deploying the model for real-time tweet classification. This work was enhanced using a Twitter API promising safety protocols on social media.

Fig. 4 Comparison of different DBN models

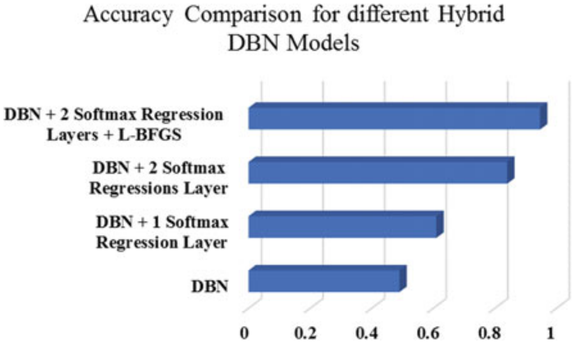


Fig. 5 Evaluation metrics

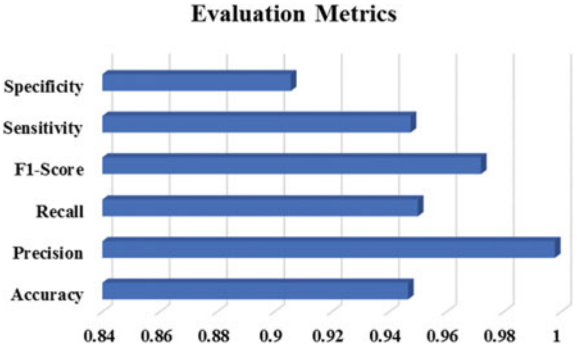


Table 1 Confusion matrix for classifying hate tweets

n = 9077	Prediction = Yes	Prediction = No
Actual = Yes	**8885**	**20**
Actual = No	**492**	**192**

4.2 Confusion Matrix

The confusion matrix is one method of displaying the percentage of suitably predicted labels for a given algorithm. A confusion matrix is a representation of findings from classification problem predictions. For example, the confusion matrix for DBN is presented in Table 1. During the testing of the model, out of 9077 tweets, a total of 8885 tweets were correctly classified as hate tweets, and 20 were classified as regular tweets. And 492 tweets were wrongly classified as hate tweets, while 192 were correctly classified as general tweets. This gives an assurance that the deep belief network can be employed to classify hate tweets. The accuracy can be improved by adapting the pre-processing of the dataset and using the state-of-the-art word embedding technique.

5 Conclusion

This work investigated a hybrid tweet classification model built using DBN and two softmax regression layers plus L-BFGS and analyzed various classification metrics. The model's accuracy is 94.66% which can be further improved by using a vast dataset or by making it real time. The neural network used has shown better accuracy and predictions with large amounts of data collected using Twitter API than other DBN models considered for this work. The future scope of this work can be extended to identifying the type of hate speech using multiple labels, which is supported by softmax regression. This model can also detect hate speech in tweets written in Indic languages by pre-processing the tweets using iNLTK, Indic NLP Library, or Stanford NLP.

Acknowledgements This work was supported by M S Ramaiah Institute of Technology, Bangalore-560054, and Visvesvaraya Technological University, Jnana Sangama, Belagavi-590018.

References

1. Liu T (2010) A novel text classification approach based on a deep belief network. In: International conference on neural information processing, pp 314–321. Springer, Berlin, Heidelberg
2. Sarikaya R, Hinton GE, Deoras A (2014) Application of deep belief networks for natural language understanding. IEEE/ACM Trans Audio Speech Lang Process 22(4):778–784
3. Koo J, Klabjan D (2020) Improved classification based on deep belief networks. In: International conference on artificial neural networks, pp 541–552. Springer, Cham
4. Zheng J, Zheng L (2019) A hybrid bidirectional recurrent convolutional neural network attention-based model for text classification. IEEE Access 7:106673–106685
5. Jiang M, Liang Y, Feng X, Fan X, Pei Z, Xue Y, Guan R (2018) Text classification based on deep belief network and softmax regression. Neural Comput Appl 29(1):61–70
6. Abdel-Zaher AM, Eldeib AM (2016) Breast cancer classification using deep belief networks. Expert Syst Appl 46:139–144
7. Zheng WL, Zhu JY, Peng Y, Lu BL (2014) EEG-based emotion classification using deep belief networks. In 2014 IEEE international conference on multimedia and expo (ICME), pp 1–6. IEEE
8. Dang NC, Moreno-García MN, De la Prieta F (2020) Sentiment analysis based on deep learning: A comparative study. Electronics 9(3):483
9. Biere S, Bhulai S, Analytics MB (2018) Hate speech detection using natural language processing techniques. Master business analytics department of mathematics faculty of science
10. Muhammad IZ, Nasrun M, Setianingsih C (2020) Hate speech detection using global vector and deep belief network algorithm. In: 2020 1st international conference on big data analytics and practices (IBDAP), pp 1–6. IEEE
11. Yepes AJ, MacKinlay A, Bedo J, Garvani R, Chen Q (2014). Deep belief networks and biomedical text categorization. In: Proceedings of the Australasian language technology association workshop 2014, pp 123–127
12. Yang Z, Pang X (2017) Research and implementation of a text classification model based on combination of DAE and DBN. In: 2017 10th international symposium on computational intelligence and design (ISCID). vol 2, pp 190–193. IEEE
13. Song J, Qin S, Zhang P (2016). Chinese text categorization based on deep belief networks. In: 2016 IEEE/ACIS 15th international conference on computer and information science (ICIS), pp 1–5. IEEE
14. Dai X, Cheng J, Gao Y, Guo S, Yang X, Xu X, Cen Y (2020) Deep belief network for feature extraction of urban artificial targets. Mathematical problems in engineering
15. Alagarsamy A, Soundar DKR (2018) A survey paper on deep belief network for big data. Int J Comput Eng Technol 9(5):161–166
16. Zhang Z, Robinson D, Tepper J (2018) Detecting hate speech on twitter using a convolution-gru based deep neural network. In: European semantic web conference, pp 745–760. Springer, Cham
17. Gambäck B, Sikdar UK (2017) Using convolutional neural networks to classify hate-speech. In: Proceedings of the first workshop on abusive language online, pp 85–90
18. Gaydhani A, Doma V, Kendre S, Bhagwat L (2018) Detecting hate speech and offensive language on twitter using machine learning: An n-gram and tfidf based approach. *arXiv preprint* arXiv:1809.08651
19. Saha P, Mathew B, Goyal P, Mukherjee A (2018) Hateminers: detecting hate speech against women. *arXiv preprint* arXiv:1812.06700
20. Aluru SS, Mathew B, Saha P, Mukherjee A (2020) Deep learning models for multilingual hate speech detection. *arXiv preprint* arXiv:2004.06465
21. Chandrika CP, Kallimani JS (2020) Classification of abusive comments using various machine learning algorithms. In: Cognitive informatics and soft computing, pp 255–262. Springer, Singapore

Generative Adversarial Network for Colorization of Mammograms

Mohil Khimani, Shiv Raj, Urvi Oza, and Pankaj Kumar

Abstract Breast cancer is one of the most frequently diagnosed cancers among women worldwide. This paper proposes an automated system that will aid the already existing methods for accurate diagnosis in the early stages of breast cancer. Our proposed approach colorizes the breast mammogram images using a generative adversarial network (GAN) model to increase the mammogram images' features. We have used the CWAS-S dataset, which provided pixel-level annotations curated by experts for all the mammogram images. For preparing the target-colored mammogram, we colored the training images with the help of the pixel labels by assigning unique colors for each label. For auto-colorization of breast mammograms, we trained a GAN model with pairs of grayscale mammograms and target-colored mammograms. Four evaluation metrics, namely RMSE, PSNR, SSIM, and FSIM, are used to assess the estimated and target color image objectively. The colored results obtained from this approach can be used for better visualization of mammogram features. It can also help achieve better results in different automated diagnosis methods like—segmentation and classification. We performed segmentation on grayscale and colored mammograms and evaluated their performance using metrics like precision, recall, DSC, and IoU. Compared to a grayscale mammogram, a colored mammogram (generated using our method) achieved an increment of 4.16% in DSC and 15.1% in IoU metric values.

Keywords Breast mammogram · Colorization · Generative adversarial network (GAN) · Segmentation

M. Khimani · S. Raj · U. Oza (✉) · P. Kumar
Dhirubhai Ambani Institute of Information and Communication Technology (DAIICT), Gandhinagar, Gujarat, India
e-mail: 201921009@daiict.ac.in

M. Khimani
e-mail: 201801416@daiict.ac.in

S. Raj
e-mail: 201801423@daiict.ac.in

P. Kumar
e-mail: pankaj_k@daiict.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_2

1 Introduction

Breast mammograms are the primary tool for breast screening and are mostly grayscale images. Doctors analyze mammograms by visual inspection. Thus, it is desirable to enhance the images in some manner that highlights the contrasting details and makes it easier to detect the lump of cells like tumors and calcification [1]. These medical images contain precious anatomical information for clinical procedures, so we propose a method to colorize them to get a better perception of images. This paper further investigates the effects on the image quality and segmentation results due to the enhancement of monochromatic mammogram images into colorized mammogram images.

In this paper, we propose a fully integrated colorization system using generative adversarial networks (GANs) that can detect and color masses simultaneously without any user intervention in a simple framework. The main advantage of GANs is that it is an unsupervised learning algorithm; therefore, it can quickly identify the regularities or patterns in input data in an unaided automatic manner and further generates the results on data based on the model. Recently, there has been advanced studies that suggest that GANs have been demonstrating promising result in the domain of colorization [2, 3]. In our proposed approach, we have built a custom generator and a discriminator as a part of our GANs architecture. The generator model learns to generate color information of a given input image. The discriminator model classifies generated images as either real (from the domain) or fake (generated). A U-Net architecture is used to perform the task of segmentation on the original gray scale and generate colored mammogram images to determine the extent of improvement in the segmentation results.

Main contribution of our paper is as follows :

- We propose an automatic breast mammogram colorization method using GAN architecture, where target images are generated using pixel-level segmentation labels.
- Based on cumulative evidence, texture information in the medical field images has been proved an essential factor in improving the performance of segmentation of the images [4, 5]. Our method preserves the texture details of the original mammogram images during the task of colorization.
- Colored version of the given grayscale image generated using simple architecture. Quality of generated images is evaluated using four metrics—RMSE (Root mean square error), PSNR (Peak signal-to-noise ratio), SSIM (Structural similarity index metric), and FSIM (Feature similarity index metric).
- We performed segmentation task on generated colored images and used IoU, dice, precision, and recall metrics to evaluate the segmentation results.

2 Related Work

Colorization techniques have been widely used for scientific illustrations, medical image processing, and old-photo colorization to enhance image features and improve visual perception.

Over the last decade, many approaches have been proposed to address colorization issues. Most early image colorization methods were primarily influenced by conventional machine learning approaches [6–8]. Previous work falls into three broad categories, which are scribble-based [9–11], transfer-based [12–16], and fully automatic colorization [17].

The fully automatic image colorization systems [17–20] have achieved high-quality performance and can generate high-quality colorized images. Deep learning has also been utilized to colorize photos [17, 21–23]. These approaches are automated and require no user intervention. Furthermore, GANs that use deep learning architectures have recently seen considerable success in colorization tasks of images [2, 24]. However, this mentioned work requires each input and targets image of training examples to be paired. Therefore, Zhu [25] proposed a CycleGAN built on pix2pix to learn the mapping without having paired training examples.

Citing the success of GANs for unpaired image colorization, Liang [26] proposed an unpaired medical image colorization system that uses a CycleGAN and introduced perceptual loss and total variation loss for achieving better performance. This colorization system produced high-quality colored breast mammogram images. However, we proposed a novel approach to generate paired breast mammogram training samples to achieve maximum visualization and accuracy in the colorized breast mammograms.

Our colorization method is also based on GANs. We have proposed a novel approach to colorizing the breast mammograms that preserve the original mammograms' texture details and uses pixel-level annotations for colorizing the training samples. The GANs architecture proposed in this paper use a custom generator (U-Net with the backbone of ResNet18) and a custom discriminator (Deep neural network (DNN) classifier) for the task of colorization. As we will demonstrate later in this paper, the colored breast mammogram images will have better segmentation than the original grayscale mammogram images.

3 Method

3.1 Data : CSAW-S

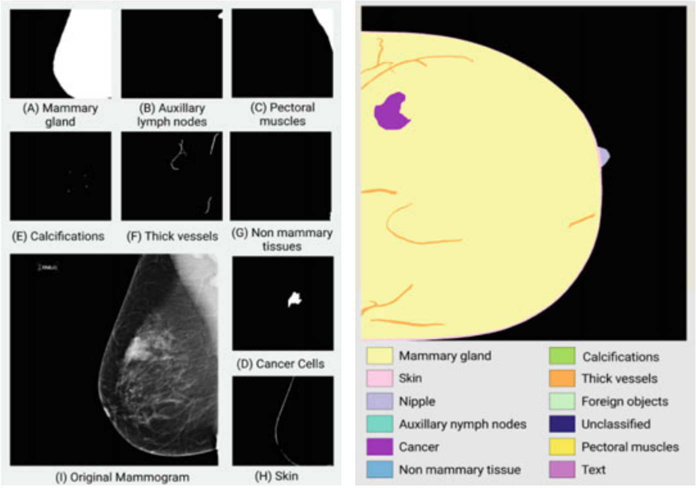
The CSAW-S dataset is a supporting subset of the CSAW dataset [27]. The CSAW-S dataset contains 172 different patient's mammography examinations along with its semantic segmentation annotations. The mammograms are split into a test set of 26 images from 23 patients and a training/validation set containing 312 images from

150 patients. The training/validation images contain pixel-level labeled annotations by an expert radiologist. The test images contain pixel-level labeled annotations from two additional radiologists [28].

3.2 Preprocessing

Preparing data: The dataset consists of folders of different patients. A folder of one patient contains the original breast mammogram of the patient and 12 different pixel-level labeled images containing the information corresponding to their respective labels. The 12 different labels for a given mammogram are—**Tumor, Calcification, Thick nerves, Nipple, Skin, Pectoral muscle, Non-Mammary tissue, Mammary glands, Foreign object, Auxillary lymph nodes, Text, and Unclassified objects.**

Figure 1a shows an example of images representing different pixel labels for a given mammogram. Here, only, eight images of different labels are shown as there was only eight labels' information available for that particular mammogram. To prepare the colored image, we first colored each of the 12 different images containing specific information of their respective labels by assigning a specific color to each of the 12 labels. After colorizing all the labeled images, they were integrated into a single picture. The colors were chosen in such a way that the important labels are in contrast with the background so that the GANs model can identify, differentiate, and



(a) Pixel level labels of mammogram image (b) Color values given to each pixel label

Fig. 1 Preparing mammogram data: **a** Tabular representation of the pixel-level labels of each patient, **b** Example of an image created by colorizing pixel-level labels and concatenating them together. The assigned color notation is also provided for reference

color them accordingly. The colors assigned to different labels are—yellow (tumor), peach (Skin), blue–purple (Nipple), Mint Majesty (Auxillary nymph nodes), purple (Cancer), Pacific Harbor (Non-mammary tissue), Conifer (Calcifications), orange (Thick vessels), Lime Dream (Foreign objects), dark blue (Unclassified), dark yellow (Pectoral muscles), light purple (Text). Figure 1b shows example of generated colored mammogram.

Augmentation: We used the augmentor package in Python to augment the training samples [29].

Images are passed through a pipeline, where each operation is applied to the grayscale and colored image together as it passes through the pipeline. Given below are the steps followed to augment images:

1. Rotating image by 90° with a probability of 0.5.
2. Rotating image by 270° with a probability of 0.5.
3. Flipping image from left to right with the probability of 0.8.
4. Flipping image from top to bottom with the probability of 0.3.

With augmentation, we were able to generate 3000 training images.

Coloring the dataset as original mammograms: In recent years, there has been accumulating evidence showing the importance of tumor texture in determining the response to cancer treatment. After assigning color values to each mammogram pixel (as shown in Fig. 1b), we want to add some more features from the original grayscale image such that spatial information about cancer or calcifications can be retained in colored mammograms when GANs predict colorized images. As colors values were assigned only based on available labels information, varying intensities of colors were required, which can help in further tasks such as segmentation and classification of mammograms.

The colored mammograms are currently in RGB colorspace and contain mixed color and intensity information about the pixels. Therefore, we used LAB color space where illumination information and color information are stored separately. The LAB colorspace has three channels; the ‘L’ channel stores lighting component; ‘A’ and ‘B’ store green–red and blue–yellow color values, respectively. The colored images are converted into LAB colorspace to extract the color components, which are ‘AB’. Then, the components are combined with the original grayscale images, which can be treated as the ‘L’ component. The diagram representation of the method mentioned above can be seen in Fig. 2. The results of the colored images obtained with the help of this method are shown in Fig. 3. After this step, we have input grayscale mammogram and target-colored mammogram pair, which are used to train GAN to predict colors based on the target-colored mammogram ‘AB’ components.

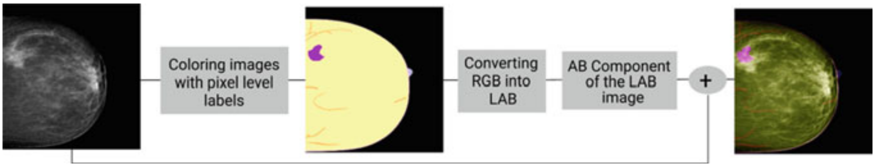

Coloring images with pixel level labels
Converting RGB into LAB
AB Component of the LAB image
+

Fig. 2 Diagram representation of colorization method: First, the grayscale images are converted into colored images using pixel labels. Its color components in LAB space are combined with grayscale images to form colored images with texture details

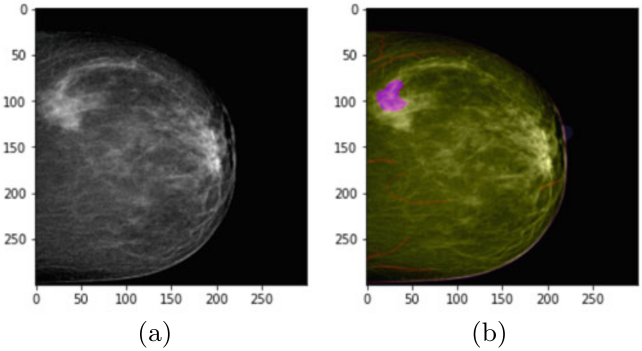


Fig. 3 Grayscale image (**a**) and Image created with grayscale and 'AB' component of pixel-labeled color image (**b**)

3.3 GAN Training

To color the mammograms, GAN architecture with a custom generator having a backbone of ResNet18 is used. The generator is a dynamic U-Net consisting of many sequential blocks followed by four U-Net blocks and one residual block. The discriminator is a combination of 4 sequential models. The diagram representation of generator and discriminator is shown in Fig. 5a, b, respectively. These two architectures together form the GAN model used for the colorization task. The generator module generates 'AB' component or artificial colors for a given grayscale mammogram, whereas the discriminator module works as a classifier. It tries to classify generated 'AB' component as real or fake. The discriminator loss penalizes the discriminator for miss-classification. Once the discriminator is trained, based on its calculated loss, backpropagation through discriminator and generator is applied; gradients are calculated, and generator weights are updated (Fig. 4).

For training our GANs model, we used a batch size of 16 images and trained the model for 75 epochs. We used Adam optimizer for training and binary cross-entropy (BCE) for loss function with logits, which is much better than the simple sigmoid function and is numerically more stable. It compares each predicted probability to actual output, i.e., 0 or 1. The loss function will then calculate the score, penalizing

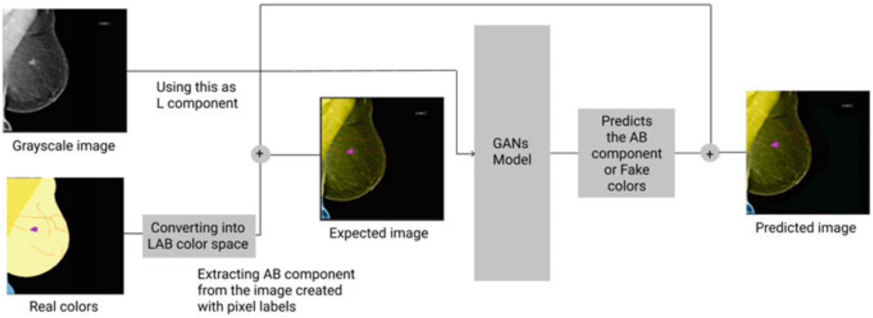
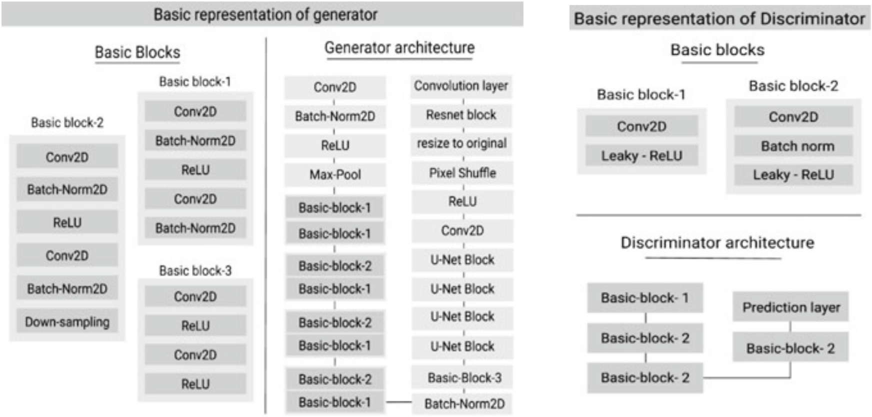

Grayscale image
Using this as L component
Real colors
Converting into LAB color space
Extracting AB component from the image created with pixel labels
Expected image
GANs Model
Predicts the AB component or Fake colors
Predicted image

Fig. 4 Diagram representation of our model: Initially, the grayscale images are converted into colored images, as shown in Fig. 2; the 'L' component of the colored image is passed through the DNN where it predicts the 'AB' component and compares the predicted 'AB' component with the 'AB' component of the original colored image



(a) Diagram representation of Generator

(b) Diagram representation of Discriminator

Fig. 5 GAN architecture used for colorization: **a** Block diagram representation of generator, **b** Block diagram representation of the discriminator

the probability based on the absolute distance from the expected value. The formula for the loss function can be represented as Eq. (1).

$$\text{BCE} = \frac{1}{N}\sum_{n=1}^{N} -\big(y_i * \log(p_i) + (1 - y_i) * \log(1 - p_i)\big) \qquad (1)$$

where y_i is the predicted value and p_i is the probability of the predicted value.

4 Results

4.1 Predicted Colored Mammogram

The predicted results by the GANs model are as shown in the Fig. 6. Results demonstrate enhanced visualization of the breast mammogram images. The model was tested on 300 images. Here, the purple-colored area is the tumor; the light yellow-colored area is the mammary gland; the dark yellow-colored area is the pectoral muscles; the dark blue-colored area is the non-mammary tissue, and light red-colored lines running through the mammary gland are thick vessels. Four different metrics, i.e., RMSE, PSNR, SSIM, and FSIM, are used to measure the model's performance. Out of the various index quality metrics (IQMs), these indices relate best to the radiologist's perception of diagnostic image quality [30]. RMSE value represents the square root of the second sample moment of differences between anticipated and observed values.

PSNR is the ratio between the maximum possible signal power and the distorting noise power.

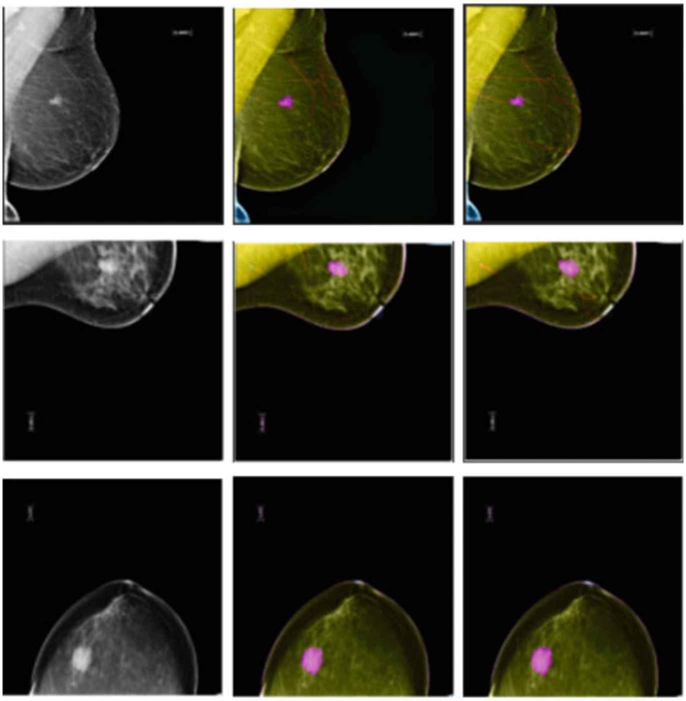


Fig. 6 Colored mammogram results: 1st column from the left is the grayscale image; 2nd column from the left is the predicted image, and 3rd column from left is the target-colored image

Table 1 Metrics used for quality assessment of colorized mammogram

Metric	Value
RMSE	2.52×10^{-7}
PSNR	104.9 dB
SSIM	0.99
FSIM	98.4

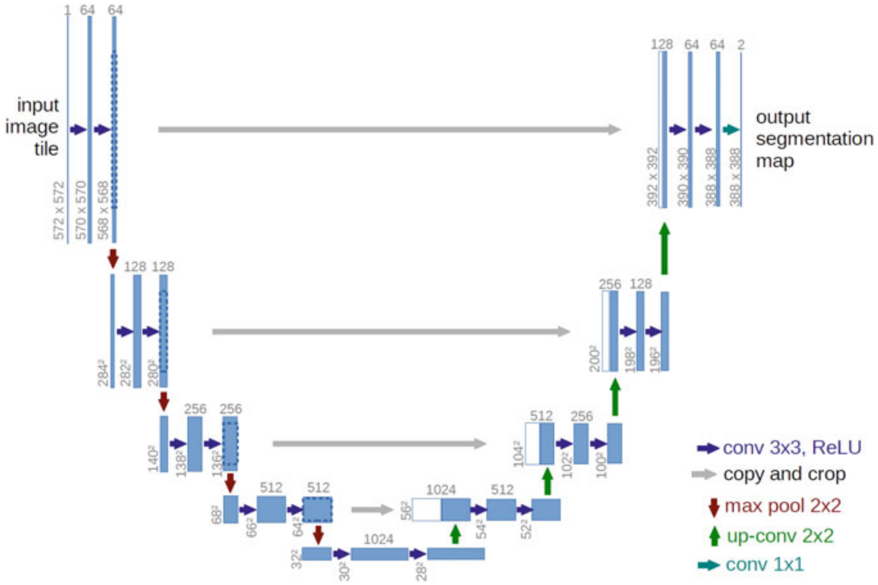


Fig. 7 U-Net architecture which was used for segmentation task [31]

SSIM metric value indicates the structural similarity, and FSIM measures the similarity of features between the generated image and the original image.

Values of these four metrics are reported in Table 1.

4.2 Segmentation Results

A colorized mammogram can help a DNN achieve better performance in many applications such as segmentation, classification, and so forth. Hence, to prove the effectiveness of the colorizing method used in this paper, segmentation results of grayscale and colored images are being compared.

We used the GANs model to generate 300 colored images and applied augmentation to both the grayscale and colored images to get 1500 images to train the

Table 2 Segmentation model results

Metric	Grayscale mammogram	Colored mammogram
DSC	0.385	**0.401**
IOU	0.331	**0.381**
Recall	0.384	**0.456**
Precision	**0.880**	0.771

segmentation models. The obtained grayscale and colored images were further subjected to a segmentation process. A U-Net architecture was used to segment these images (architecture is shown in the Fig. 7). Two different models with the same architecture (One for gray scale and one for colored images) were trained to compare the results. Both were given two inputs, colored/grayscale images and pixel-level labeled tumor masks—the model generates a mask which was used to evaluate the segmented results.

Four metrics were used to evaluate the segmentation results, i.e., dice score (DSC), Intersection over Union (IoU), recall, and precision. IoU is a ratio of the area of overlap between the predicted image and ground-truth image and the area of union; it ranges from 0 to 1. Hence, an increase in IoU score means an increase in the area of overlap, which leads to an increase in accuracy. Dice score is the ratio of twice the area of overlap between predicted and ground-truth image and the total number of pixels. Dice and IoU are much similar and positively correlated, So, including just one of them might be enough, but in some instances, IoU has the upper hand because it always tends to calculate the worst-case performance. In contrast, dice tends to calculate average-case performance. So, to keep a balance, both metrics are being included. The segmentation results for the ground-truth grayscale mammogram samples and predicted colored mammogram samples are reported in Table 2. As evident from the results, the predicted colored images produced better segmentation results than the ground-truth images. Both the dice and IoU metric values demonstrated a significant increase in their values. Increased recall value indicates low false-negative rates. However, segmentation results on colored mammograms have a low precision score compared to the original mammogram, which indicates an increased number of false positives on colored mammograms.

$$\text{precision} = \frac{\text{True positive}}{\text{True positive} + \text{False positive}} \tag{2}$$

$$\text{Recall} = \frac{\text{True positive}}{\text{True positive} + \text{False negative}} \tag{3}$$

5 Conclusion

The purpose of this research was to provide a productive way by which breast mammograms can be colored, so the abnormalities or any other aberrant behavior visible in the mammograms can be detected and treated readily. We did this successfully with the help of the CSAW-S dataset, which contained pixel-level labels for 12 different classes. With our approach, we were able to color and detect tumors and calcifications. This method colors not only tumor and calcification but also nine other anatomical classes, which can prove to be helpful for other future research. We used GANs architecture which is dexterous when it comes to creating authentic images. GANs are often used for colorizing natural images, and as we can infer from the results, it shows favorable outcomes even for medical images colorization. The predicted colored images demonstrated favorable results when they were subjected to segmentation. This increased performance in the segmentation task indicates enhancement of features due to colorization of the images.

References

1. Tien Pham W, Tat Pham T, Illescas Maldonado P (2020) Enhancing mammogram images with segmentation and colorization for assisting breast cancer detection
2. Dhir R, Ashok M, Gite S, Kotecha K (2020, December) Automatic image colorization using GANs. In: International conference on soft computing and its engineering applications. Springer, Singapore, pp 15–26
3. Liang Y, Lee D, Li Y, Shin BS (2021) Unpaired medical image colorization using generative adversarial network. Multimedia Tools Appl 1–15
4. Mughal B, Muhammad N, Sharif M (2018) Deviation analysis for texture segmentation of breast lesions in mammographic images. Euro Phys J Plus 133(11):1–15
5. Anjaiah P, Prasad KR, Raghavendra C (2018) Effective texture features for segmented mammogram images. Int J Eng Technol 7(3):666–669
6. Huang YC, Tung YS, Chen JC, Wang SW, Wu JL (2005, November) An adaptive edge detection based colorization algorithm and its applications. In: Proceedings of the 13th annual ACM international conference on Multimedia, pp 351–354
7. Yatziv L, Sapiro G (2006) Fast image and video colorization using chrominance blending. IEEE Trans Image Process 15(5):1120–1129
8. Luan Q, Wen F, Cohen-Or D, Liang L, Xu YQ, Shum HY (2007, June) Natural image colorization. In: Proceedings of the 18th Eurographics conference on rendering techniques, pp 309–320
9. Lagodzinski P, Smolka B (2009, October) Colorization of medical images. In: Proceedings: APSIPA ASC 2009: Asia-Pacific signal and information processing association, 2009 annual summit and conference, pp 769–772. International Organizing Committee
10. Furusawa C, Hiroshiba K, Ogaki K, Odagiri Y (2017) Comicolorization: semi-automatic manga colorization. In: SIGGRAPH Asia 2017 technical briefs, pp 1–4
11. Taigman Y, Polyak A, Wolf L (2016) Unsupervised cross-domain image generation. arXiv preprint arXiv:1611.02200
12. Li B, Lai YK, John M, Rosin PL (2019) Automatic example-based image colorization using location-aware cross-scale matching. IEEE Trans Image Process 28(9):4606–4619

13. Charpiat G, Hofmann M, Schölkopf B (2008, October) Automatic image colorization via multimodal predictions. In: European conference on computer vision. Springer, Berlin, Heidelberg, pp 126–139
14. Chia AYS, Zhuo S, Gupta RK, Tai YW, Cho SY, Tan P, Lin S (2011) Semantic colorization with internet images. ACM Trans Graph (TOG) 30(6):1–8
15. Morimoto Y, Taguchi Y, Naemura T (2009) Automatic colorization of grayscale images using multiple images on the web. In: SIGGRAPH 2009: talks, pp 1
16. Welsh T, Ashikhmin M, Mueller K (2002, July) Transferring color to greyscale images. In: Proceedings of the 29th annual conference on computer graphics and interactive techniques, pp 277–280
17. Larsson G, Maire M, Shakhnarovich G (2016, October) Learning representations for automatic colorization. In: European conference on computer vision. Springer, Cham, pp 577–593
18. Joshi MR, Nkenyereye L, Joshi GP, Islam SM, Abdullah-Al-Wadud M, Shrestha S (2020) Auto-colorization of historical images using deep convolutional neural networks. Mathematics 8(12):2258
19. Zhang R, Isola P, Efros AA (2016, October) Colorful image colorization. In: European conference on computer vision. Springer, Cham, pp 649–666
20. Cheng Z, Yang Q, Sheng B (2015) Deep colorization. In: Proceedings of the IEEE international conference on computer vision, pp 415–423
21. Cheng Z, Yang Q, Sheng B (2017) Colorization using neural network ensemble. IEEE Trans Image Process 26(11):5491–5505
22. Iizuka S, Simo-Serra E, Ishikawa H (2016) Let there be color! Joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. ACM Trans Graph (ToG) 35(4):1–11
23. He M, Chen D, Liao J, Sander PV, Yuan L (2018) Deep exemplar-based colorization. ACM Trans Graph (TOG) 37(4):1–16
24. Nazeri K, Ng E, Ebrahimi M (2018, July) Image colorization using generative adversarial networks. In: International conference on articulated motion and deformable objects. Springer, Cham, pp 85–94
25. Zhu JY, Park T, Isola P, Efros AA (2017) Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE international conference on computer vision, pp 2223–2232
26. Liang Y, Lee D, Li Y, Shin BS (2021) Unpaired medical image colorization using generative adversarial network. Multimedia Tools Appl, 1–15
27. Dembrower K, Lindholm P, Strand F (2020) A multi-million mammography image dataset and population-based screening cohort for the training and evaluation of deep neural networks-the cohort of screen-aged women (CSAW). J Digital Imaging 33(2):408–413
28. Matsoukas C, Hernandez AB, Liu Y, Dembrower K, Miranda G, Konuk E, Haslum JF, Zouzos A, Lindholm P, Strand F, Smith K (2020, November) Adding seemingly uninformative labels helps in low data regimes. In: International conference on machine learning. PMLR, pp 6775–6784
29. Bloice MD, Stocker C, Holzinger A (2017) Augmentor: an image augmentation library for machine learning. arXiv preprint arXiv:1708.04680
30. Mason A, Rioux J, Clarke SE, Costa A, Schmidt M, Keough V, Beyea S (2019) Comparison of objective image quality metrics to expert radiologists' scoring of diagnostic quality of MR images. IEEE Trans Med Imaging 39(4):1064–1072
31. Ronneberger O, Fischer P, Brox T (2015, October) U-net: convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention. Springer, Cham, pp 234–241

Coverage, Capacity and Cost Analysis of 4G-LTE and 5G Networks

A Case Study of Ahmedabad and Gandhinagar

Jay Gandhi and Zunnun Narmawala

Abstract The prediction of future ten years shows exponential growth in wireless connections, high-speed Internet and data demands. Mobile network operators (MNOs) need to provide essential capacity for more than 100 billion connections in the global mobile communications network. The contribution of this paper is the analysis of the coverage, capacity and cost requirement of 4G-LTE and 5G networks across the Ahmedabad and Gandhinagar cities for the period of 2019–2029. We forecast the number of 4G-LTE and 5G subscribers and their data demands over the years. To accomplish such capacity requirement and to ensure the coverage across the cities, various radio propagation models are used to calculate the base station site requirement. The published data on networks deployment cost for various scenario (rural, urban, suburban, dense urban) are used to evaluate capital expenditure (CAPEX) and operational expenditure (OPEX) cost. The key findings of the study are as follows: (a) 4G-LTE is insufficient to provide high-speed data without acquiring additional spectrum bandwidth (b) 5G technologies can provide significant coverage and capacity even in the dense urban area (c) The cost of deploying 5G infrastructure is almost three times higher than the 4G-LTE.

Keywords 4G-LTE · 5G · Coverage and capacity · Cost modeling · CAPEX · OPEX

1 Introduction

The number of connected devices is expected to grow in future, with an approximation of more than 70 billion devices by 2025 [17, 18]. There is an outburst in the demand of high-speed data due to bandwidth-demanding applications such as augmented and virtual reality (ARVR), gaming, video streaming and mobile computing [13]. User expectations are growing for improved service in terms of higher data rate, greater

J. Gandhi (✉) · Z. Narmawala
Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad, India
e-mail: jaygandhi7591@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_3

system capacity, and reduced latency. The wireless communication is the backbone for Smart Cities, Internet of Things (IoTs), Smart Grids, Smart Healthcare, Smart Infrastructure, and Smart Devices [17]. Without adequate and fair connectivity in wireless communication, it is not conceivable to leverage the benefits associated with these applications. The article analyzes the future coverage, capacity and cost requirements for 4G and 5G networks in urban areas for adequate connectivity. We have chosen the real-world scenario of Ahmedabad and Gandhinagar cities located in Gujarat, India as a case study.

The fourth-generation (4G) cellular network uses the long-term evolution (LTE) networks to facilitate high-speed wireless communication to the users. LTE is a novel cellular technology that significantly increases the capacity and speed of mobile networks as compared to 3G cellular networks [2]. To leverage the maximum benefits of 4G networks, the MNOs need to obtain adequate radio spectrum bandwidth to meet user demands. The spectrum band is also known as the carrier frequencies (CFs) which are utilized to provide the 4G services. The selection of appropriate CFs according to the target area is essential to provide the most reliable coverage in that region. To determine the capacity requirement of particular region, MNOs need to forecast the number of subscribers with their potential usage of networks. In all, to assess technical feasibility for 4G-infrastructure deployment, we need to analyze scenario (rural, urban, suburban, and dense urban), available spectrum, subscriber forecast, number of base stations (BTS) required and estimated cost. With 5G, an extensive amount of data can be transmitted more efficiently than 4G-LTE. 5G is designed to enhance the scope of wireless communication surpassing the capabilities of 4G-LTE. 5G has the potential to manage millions of devices at rapid speed and also to extend accessibility [5]. Various usage scenario such as Enhanced Mobile Broadband (eMBB), Ultra-Reliable and Low-Latency Communications (URLLC) and Massive Machine-type Communications (mMTC) are proposed in 5G. 5G was introduced to improve diverse factors such as speed, coverage, capacity and traffic density [4]. Generally, user experienced the data rate of 10 Mbps for 4G whereas rate quoted for 5G is in the range of 1–10 Gbps and beyond. MNOs need to figure out following important questions: (i) How to achieve such high data rate and meet users' demands for different usage scenario? (ii) How much will it cost to deploy the 5G networks? (iii) What will be the cost per user?

The major contributions of this study are as follows:

1. The study forecasts the number of subscribers in Ahmedabad and Gandhinagar for 2019–2029 duration.
2. Our study performs a detailed analysis of the achievable coverage area of 4G-LTE and 5G networks for diverse deployment scenarios.
3. Based on user data demand, the study calculates an adequate number of sites needed in a given year to meet the capacity requirement.
4. The study performs a strategic analysis of 4G and 5G to understand the investment cost involved in a specific scenario. The study identifies challenges faced by society in terms of coverage, capacity and helps in decision making when it comes to wireless communication.

The remainder of the paper is organized as follows: Sect. 2 gives a concise summary of the relevant work. A comprehensive description of the 4G and 5G analysis is provided in Sects. 3 and 4, respectively. Section 5 presents results and discussion. Section 6 puts forth the concluding remarks and directions for future work.

2 Related Work

This section considers the theoretical and technical aspects along with the research focusing on their coverage, capacity and cost modeling. Several studies have investigated the coverage and capacity requirements for 4G-LTE and 5G infrastructure to meet the demands of contemporary users for the specific region or deployment scenario. This section presents the literature survey of the papers that closely resembles our study.

In [8], the authors have evaluated the technical feasibility of deploying 4G-LTE networks by considering the 22 telecom circles (TCs) of India. The study estimates the number of 4G-LTE sites needed to satisfy the requirement of coverage and capacity for the years 2016–2026. Using the Okumura-Hata and COST-231 radio propagation models, the study assesses the infrastructure requirements over the years for the major cities and states of India. The study shows that India will be "Capacity Constrained" with available spectrum bandwidth. Also, coverage and capacity requirements are highest in the metro cities.

In [7], the authors have demonstrated the effectiveness of the newly revealed 700 MHz band as compared to 1800 and 2100 MHz bands. The study attempts to analyze the financial profitability of 700 MHz for providing 4G-LTE service across India. The results show that the 700 MHz band gives better coverage, capacity and cost-effective solution. The comparative analysis shows that the 700 MHz band has ample potential for financial recoverability for the mobile network operators (MNOs).

In [9], a study presented 4G-LTE network planning to determine the number of sites required for Banepa city. The 1800 MHz band and 5 MHz bandwidth are utilized to evaluate the coverage and capacity estimation in the area. The simulation result obtained using Atoll software shows that 92.6% area of Banepa is covered by deploying seven sites. The uncovered area is caused due to co-channel interference. The deployed sites achieve the capacity of 33 Mbps per cell.

In [17], the study was undertaken to measure the coverage, capacity and cost of various 5G infrastructure strategies for the Netherlands. The evaluated strategies are the spectrum integration approach which uses 700, 1500 MHz, 3.5 GHz bands and the small cell strategy. It estimates the capacity provided to the users by the existing spectrum and intimates at which stage small cell deployment is required. The analysis is performed to achieve the per user speed of 30, 100, or 300 Mbps and to propose investment strategies to accomplish such demand. The study shows a 40% improvement in per user traffic capacity than the existing 4G-LTE capacity.

In [16], the study focused on providing ultrafast 50 Mbps speed and analyzed inference in Britain. The baseline scenario to analyze the deployment is in terms of coverage, infrastructure sharing, end-user speed and cost. The spectrum integration approach is similar as discussed in [17]. The analysis shows that 90% population will be covered by 5G by 2027 with 50 Mbps speed. But, the remaining 10% coverage increases cost exponentially. Moreover, the spectrum plays a significant role in decreasing the cost of a network.

In [19], the authors have analyzed 5G deployment for the dense urban area of London. To achieve the 100 Mbps speed everywhere, networks were developed using a spectrum band of 700 MHz, 3.5 GHz and 24–27.5 GHz. The result shows that 700 MHz is an exceptional band for coverage but didn't achieve 100 Mbps speed at every place. However, 3.5 GHz and 24–27.5 GHz bands can achieve the targeted speed, but the deployment cost increases 4–5 times as compared to the 700 MHz band.

In general, the existing study shows an inadequate techno-economic estimation of 4G-LTE and 5G. Our study shows the comparative analysis of 4G-LTE and 5G by considering various realistic scenarios. It helps the researcher to develop cost-effective and high-speed wireless infrastructure in future.

3 Analysis of 4G-LTE

In this section, we describe the theoretical and mathematical concepts used for 4G-LTE analysis. We provide the description of the proposed strategy for subscriber forecast, coverage modeling, capacity modeling, and cost module.

3.1 Forecasting Subscribers Using Bass Diffusion Model

The Bass model is widely used for new products sales forecasting and technology forecasting [12]. It has been tested by industries for many novel technologies and products. Using the Bass diffusion model, the 4G-LTE and 5G subscribers are divided into two categories, namely innovators and imitators [8]. The innovators are usually independent and don't rely on other subscribers for making decisions. Whereas, imitators are led by other subscribers and gather the information using unconventional sources. For a particular product, mainly, three Bass model parameters are expressed: M-the potential market, p-coefficient of innovation, and q-coefficient of imitation. The Bass model equations (1), (2) and (3) to forecast the subscribers are formulated below:

$$N(t_i) = b_1 + b_2 N(t_{i-1}) + b_3 (N(t_{i-1}))^2 \tag{1}$$

where $N(t_i)$ is number of adoptions for ith time period and b_1, b_2 and b_3 are the Bass model coefficients.

$$M = \frac{-b_1 - \sqrt{b_1^2 - 4b_2b_3}}{2b_3}, p = \frac{b_1}{M}, q = -Mb_3 \quad (2)$$

$$F(t) = M\left[\frac{1 - e^{-(p+q)t}}{1 + \frac{q}{p}e^{-(p+q)t}}\right] \quad (3)$$

where $F(t)$ is the fraction of market potential for the time t, p and q are the coefficients of innovation and imitation respectively.

3.2 Coverage Modeling of 4G-LTE

We have applied two radio propagation models to determine the coverage area and cell range. First, the Okumura-Hata model [8] is used for carrier frequencies (CFs) of 700, 850, 1800, and 2100 MHz. Second, the COST-231 model [8] is used for CFs of 2300 and 2600 MHz. However, we need to determine maximum allowable signal attenuation (MAPL) for link budgeting and that value is used in the radio propagation model. The formulation of calculating MAPL is shown in Eq. (4).

$$\text{MAPL} = \text{EIRP}_{Tx} - \text{REC}_{\text{sen}} - \text{IM} - C_{\text{loss}} + \text{RA}_{Gx} - M + S_{\text{gain}} \quad (4)$$

where EIRP_{Tx} is the Transmitter's Equivalent Isotropically Radiated Power, REC_{sen} is receiver sensitivity, IM is the interference margin, C_{loss} is cable loss, RA_{Gx} is receiver antenna gain, M is margin of fast-fade and S_{gain} is soft handover gain. The Okumara-Hata model is formulated as:

$$d = 10^{((\text{MAPL} - A + h_{\text{ms}} - A_C)/B)} \quad (5)$$

$$A = 69.55 + 26.16 \log_{10}(f_c) - 13.82 \log_{10}(ht_b) \quad (6)$$

$$B = 44.9 - 6.55 \log_{10}(ht_b) \quad (7)$$

where d = cell radius h_{ms} and A_C values vary depending on the scenario, ht_b = height of base station antenna and f_c = CFs (MHz).

Whereas, COST-231 model is formulated as:

$$d = 10^{((\text{MAPL} - A + h_{\text{ms}} - A_C - C_M)/B)} \quad (8)$$

$$A = 46.3 + 33.9 \log_{10}(f_c) - 13.82 \log_{10}(ht_b) \quad (9)$$

$$B = 44.9 - 6.55 \log_{10}(ht_b) \quad (10)$$

$$\text{ToS} = {}^{\text{Area}}\!/_{(5.07 \times d^2)} \quad (11)$$

where C_M is 0 dB for rural, urban, and suburban and 3 dB for dense urban. The total number of sites required for specific area is calculated using Eq. (11) in which d is obtained from Eqs. (5) and (8).

3.3 Capacity Modeling of 4G-LTE

The capacity modeling is to evaluate the number of sites required to meet the subscriber demand in the specific area. Along with the number of subscribers calculated using Bass diffusion model, we need to evaluate achievable throughput of 4G-LTE site. The cell dimensioning method described in [6] is used to evaluate cellular throughput (Th) per 4G-LTE site (in Gb/site/year) which is expressed in Eq. (12). The total number of sites (CAP) required for an area to meet the capacity in a given year is evaluated using Eq. (13).

$$\mathrm{Th} = n_{\mathrm{sector}} \times \mathrm{bh}_l \times B \times \mathrm{sp}_e \tag{12}$$

where, n_{sector} = antenna sector per site, bh_l = average busy hour, B = bandwidth and sp_e = spectral efficiency of 4G-LTE.

$$\mathrm{CAP} = \frac{0.0758 \times b_h \times N_{\mathrm{SF,Year}} \times D_{\mathrm{SF,year}}}{\mathrm{Th}} \tag{13}$$

where b_h = daily busy hours traffic, $N_{\mathrm{SF,year}}$ = subscriber forecast in the given year and $D_{\mathrm{SF,year}}$ = data demand per subscriber in the given year.

3.4 Cost Modeling of 4G-LTE

The 4G-LTE networks' cost mainly depends on CAPEX and OPEX. The CAPEX is the necessary cost required to deploy infrastructure and the license cost [7]. Whereas, OPEX is the operational and maintenance cost to run the business.

$$\mathrm{CAPEX} = (\mathrm{BS}_{i,\mathrm{year}} \times \mathrm{NC}_i) + \mathrm{SPR}_{\mathrm{lic}} \tag{14}$$

$$\mathrm{OPEX} = (\mathrm{BS}_{\mathrm{i,year}} \times \mathrm{SC}_i) + \mathrm{SPR}_A \tag{15}$$

where $\mathrm{BS}_{\mathrm{i,year}}$ = number of sites in the given year, NC_i = network cost per site, $\mathrm{SPR}_{\mathrm{lic}}$ = installment of annual spectrum license, SC_i = site related expenses and SPR_A = annual spectrum charge. As per [7], we have considered spectrum license charge as 10% of base price with total 18 equal annual spectrum license installments. Also, 3.78% rate of inflation and 3% spectrum usage charge are considered based on annual revenue. Revenue is estimated from average revenue per user (ARPU), total number of subscribers and access charge.

4 Analysis of 5G

In this section, we describe the theoretical and mathematical concepts used for 5G analysis. We provide the description of the proposed strategy for subscriber forecast, coverage modeling, capacity modeling, and cost modeling.

4.1 Coverage Modeling of 5G

The 5G Network is modeled for 700 MHz, 3.5 GHz, and 24–27.5 GHz frequencies. The transmission power and receiver sensitivity parameter used to evaluate link budget is taken from [1]. The Okumura-Hata propagation model is used for 700 MHz (Macrocells), and the model obtained in [1, 3] is used for 3.5 GHz (Microcells) and 24–27.5 GHz (Hotspots), respectively. To calculate the network coverage, we need to evaluate signal-to-noise and interference ratio (SINR) for each cell. Once SINR is obtained, it is used in modified Shannon formula shown in Eq. (16) to calculate the throughput.

$$\mathrm{Th} = W \times \mathrm{Min}(\log_2(1 + 10^{(\mathrm{SINR}-\delta)/10}), \mathrm{spr}_{\max}) \tag{16}$$

where Th is the throughput in bits/sec, W is the channel width in Hz, $\mathrm{spr}_{\max}$ is spectral efficiency. The loss factor (δ) is 1.6 dB for the lower frequency band and 3 dB for the higher frequency band. Based on [11], we model the coverage of base station for the specific cell area. The base station site coverage characterized by class i for cell range ri is as follows:

$$\mathrm{Cell}_r = \pi \times rc_i^2 \tag{17}$$

According to [6, 10], the cell range varies from 0.6 to 1.4 km in urban areas with the wall penetration loss 20 dB. As per [13, 14], we have considered the range of 0.57–0.1 km for macrocell, microcell, and hotspots.

4.2 Capacity Modeling of 5G

Based on the population data and area in km^2, we need to evaluate the population density of specific area. The estimated total number of subscribers are utilized to calculate the demand as shown in Eq. (18) as proposed in [17].

$$\mathrm{Capa}_{\mathrm{sub}} = \frac{((\mathrm{den}_p) \times (\frac{\mathrm{adop}}{100}))(\frac{\mathrm{market}_{\mathrm{sr}}}{100}) \times \mathrm{sp}_{\mathrm{sub}}}{\mathrm{OBF}} \tag{18}$$

The demand estimated is Mbps per km^2 for the specific area by considering expected per user speed $\mathrm{sp}_{\mathrm{sub}}$. Overbooking factor (OBF) denotes proportion of users using the network at a time. As per [15], the system capacity ($\mathrm{sys}_{\mathrm{cap}}$) model is as follows:

$$\mathrm{Sys}_{\mathrm{cap}} = W \times \mathrm{BS}_{\mathrm{site}} \times N_{\mathrm{cell}} \times \mathrm{spr}_{\mathrm{eff}} \tag{19}$$

where W is bandwidth in MHz, $\mathrm{BS}_{\mathrm{site}}$ is total number of sites, N_{cell} is number of cells, and $\mathrm{Spr}_{\mathrm{eff}}$ is cell spectral efficiency. The average spectral efficiency varies between 3.8 and 6.6 bits/sec/Hz/cell in the urban environment [15]. The demand and system capacity is then utilized to calculate the number of 5G sites required for the specific area.

4.3 Cost Modeling of 5G

The study provides the cost estimation of 5G networks using wide range of sources. We have considered the openly published data for calculating the CAPEX and OPEX cost in 5G networks [19]. The deployment and operational cost of 5G is divided into six major areas: (1) Initial deployment cost of site (2) Backhaul (3) Site Rental (4) Maintenance and Operations (5) Spectrum Cost and (6) Power Cost. The total cost of ownership (TCO) is derived with the assumption of CAPEX spent in 1 year, and OPEX is discounted 5% each year over the 10 year period [15]. From the coverage and capacity analysis, we evaluate the number of sites required for specific area. Now, it is very straightforward to calculate cost, based on base station density. We measure the total cost of 5G networks to cover entire area by considering expected demand of subscribers.

5 Results

This section details the results of coverage, capacity and cost analysis of 4G-LTE and 5G Networks. The analysis is undertaken using the case study of Ahmedabad and Gandhinagar City. Ahmedabad is the largest city of the Gujarat state whereas Gandhinagar is the capital of the Gujarat state. The population of Ahmedabad is around 80 L in 2020 which is growing at the rate of 2.54% annually. Whereas, Gandhinagar population is around 3.4 L, and growth rate is 4.15% annually. The population density is approximately 9900 and 1700 persons per square kilometer of Ahmedabad and Gandhinagar, respectively. By 2030, there could be more than 1 crore people residing in Ahmedabad and around 5 L in Gandhinagar.

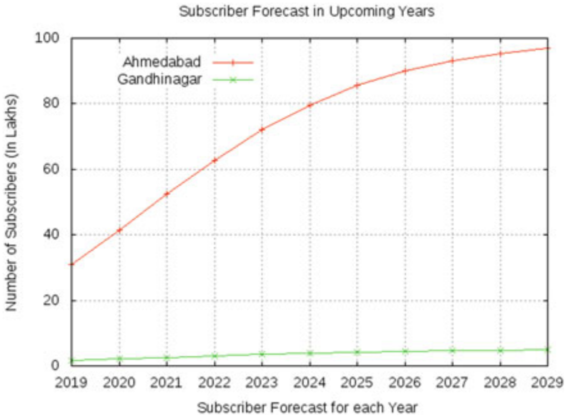


Fig. 1 Subscriber forecast of 4G-LTE and 5G for the period of 10 Years

5.1 Subscriber Forecast for 4G-LTE and 5G

Using Bass diffusion model, we have predicted the subscription trend for 4G-LTE and 5G networks from the year 2019–2029. We have used the historical data of 4G-LTE adoption and cellular subscriber data [18] for the ultimate market potential values. The corresponding assumptions are made for the city Ahmedabad and Gandhinagar. Depending on the terrain cluster of cities, the empirical propagation models can be used to design and planning of wireless networks for the other cities. For the analysis of 5G networks, it is assumed that the every 4G-LTE subscriber is potential adopter of 5G networks in the future. Figure 1 illustrates the subscriber forecast for Ahmedabad and Gandhinagar. We observe that subscription will increase significantly over the next 10 years, especially in Ahmedabad. It indicates that substantial investment in cellular infrastructure is required for upcoming 10 years to meet the subscribers' demand.

5.2 4G-LTE Coverage Modeling

For the 4G-LTE coverage modeling, the input assumptions for radio propagation model and link budget calculation have been taken from [7, 8]. The radio propagation model is evaluated for different carrier frequencies (CFs) to calculate the total number of 4G-LTE sites required for full coverage. The analysis is performed using 700, 850, 1800 and 2100, 2300, and 2600 MHz CFs. The frequencies vary in terms of cellular radius for the scenario such as dense urban, urban, suburban, and rural. Based on the population density, we have considered Ahmedabad and Gandhinagar as a dense urban and urban area, respectively. Figure 2 provides the details about the number of sites required with various CFs. We can infer that lower frequency bands offer significantly higher coverage. Also, significantly, higher number of sites are required for Ahmedabad which is dense urban area compared to Gandhinagar.

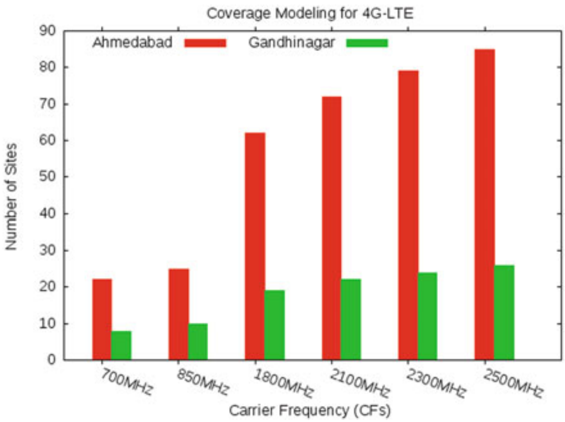


Fig. 2 Number of 4G-LTE sites for entire city coverage

5.3 4G-LTE Capacity and Cost Modeling

The number of 4G-LTE sites required to meet the capacity is evaluated after measuring number of subscribers, annual data demand per subscriber, and data volume capabilities per site. The average data usage per month in Gujarat is about 13 Gb which is higher than national average of 10 Gb. Also, it is assumed that demand increases exponentially over the years [18]. Based on subscribers' forecast in Sect. 3 and its data demand, we measured the data volume demand for each city. The data volume capabilities per site are measured using Eq. (14). Finally, the number of sites required to meet the demand in each year is calculated using Eq. (15). Figure 3 shows results for Ahmedabad and Gandhinagar. The results indicate that 4G-LTE can meet the coverage requirement, but huge infrastructure deployment is required to meet the capacity requirement of densely populated cities like Ahmedabad. To ensure capacity, the MNOs will have to deploy significantly more 4G-LTE infrastructure in upcoming years.

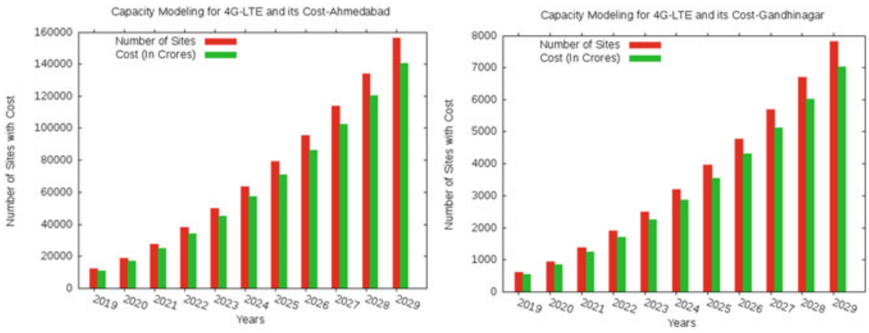


Fig. 3 Number of 4G-LTE sites required for meeting capacity and its cost

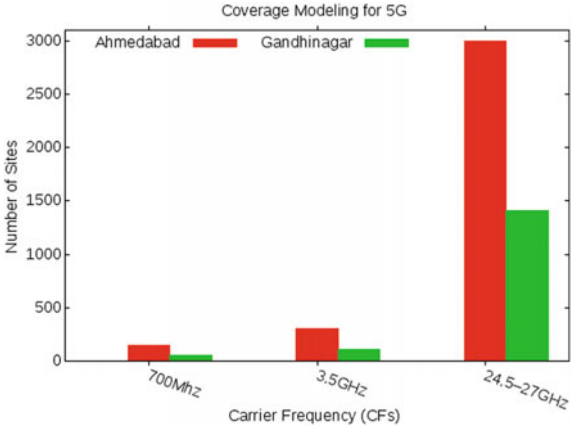


Fig. 4 Number of 5G-LTE sites for entire city coverage

The investment in 4G-LTE infrastructure comprises of site installation, backhaul, license cost, operation and maintenance cost. The base station installation is estimated around ₹ 20–22 L, and site build out and equipment will be over ₹ 24–26 L per site. The operational cost of power supply and maintenance is estimated around ₹ 5–6 L per site whereas ₹ 12–14 L for site lease and leased line. To calculate the spectrum cost, the base price of each band is taken from spectrum auction of TRAI. The deployment cost relatively depends on the number of subscribers per site. The CAPEX and OPEX per subscriber are relatively higher in the low-density area. The cost depends on the network utilization. If data usage is higher due to either number of subscribers or per subscriber's demand, then the cost per user can be around 0.9–1.2 K/month. Along with the number of sites, Fig. 3 also shows the cost of deployment for Ahmedabad and Gandhinagar cities.

5.4 5G Coverage Modeling

The results of 5G coverage obtained in this section are for 700 MHz, 3.5 GHz, and 24–27.5 GHz CFs. For 700 MHz frequency, the higher site density does not improve the performance due to small bandwidth. The 3.5 GHz frequency provides rapid speed and adequate coverage for outdoor compared to indoor. However, the higher frequency band of 24–27.5 GHz can only be modeled for outdoor as it does not penetrate through walls and windows easily. Based on the evaluation, Fig. 4 shows the number of 5G sites required for the coverage of Ahmedabad and Gandhinagar City.

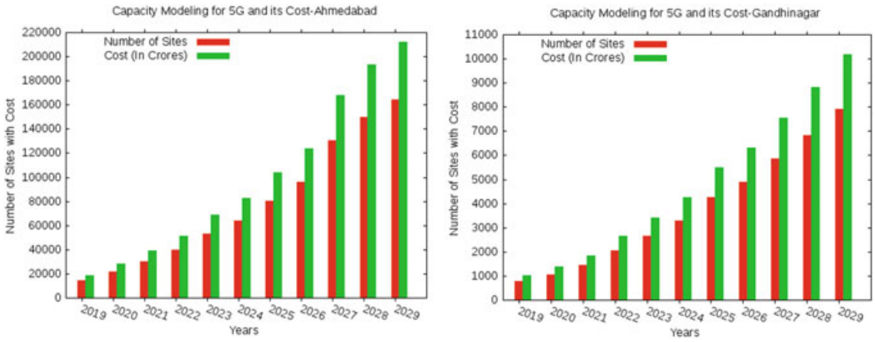


Fig. 5 Number of 5G-LTE sites required for meeting capacity and its cost

5.5 5G Capacity and Cost Modeling

The analysis is made to provide the user-experienced data rate of 100 Mbps to each subscriber. We assume that average usage is 100 Gb/month which varies between 70 and 120 Gb per month. Due to lower bandwidth and interference, 700 MHz frequency does not provide much capacity. Whereas, 3.5 GHz and 24.5–27 GHz provide significant capacity of 40–120 Gbps/km^2. The wall loss plays an important factor because the high-frequency band does not penetrate through wall. Figure 5 shows the number of 5G sites required to achieve the required capacity. We have determined the annual cost of sites based on the number of sites required to ensure full capacity in the city. As discussed in Sect. 4, cost of deploying of 5G infrastructure depends on various things which are as follows: For CAPEX, the cost is between ₹ 34–35 L for equipment, ₹ 39–40 L for site build out, and ₹ 23–24 L for installation. For OPEX, the cost ranges from ₹ 4–6 L for operation and maintenance, ₹ 11–14 L for site lease and leased line, ₹ 4–7 L for the electric power. It is important to take a note that spectrum cost varies in different countries and regulatory regimes. We assume the spectrum cost between 1 and 1.5 K/MHz/km^2. Subsequently, the use of the new spectrum generates a need for advancing radio interface and antenna to enhance efficiency. The analysis shows that the average cost per user ranges between 2.8 and 3 K/month for the high network utilization scenario. Figure 5 shows the cost of deployment of 5G for Ahmedabad and Gandhinagar.

6 Conclusion

The ultimate objective of the study is to evaluate the feasibility of deploying 4G-LTE and 5G across the Ahmedabad and Gandhinagar. The paper estimates the infrastructure required to ensure competent coverage and capacity based on the number of subscribers and their data demand for the period of 2019–2029. There is huge growth in the 4G subscribers after the availability of 4G devices. Also, the average data usage increased by up to 55% from 2018 to 2020. For 4G-LTE, we observe that infrastructure demand is much higher in densely population area. The coverage and capacity requirement determines that in densely populated Indian cities, capacity requirement is dominating factor in determining infrastructure need in comparison to coverage requirement. Also, the analysis exhibits that the 700 MHz is cost-effective frequency band with a remarkable potential of profitability for 4G-LTE. The successor of 4G-LTE, 5G is expected to deliver high-speed data by utilizing the high-frequency bands. The higher frequency bands in 5G achieves adequate outdoor coverage, but for indoor coverage, we need to locate micro-base stations within the building to provide significant capacity thus requiring additional 5G sites. Even if we consider the least deployment cost of 5G, still, it is three times more expensive than the 4G-LTE.

References

1. Akdeniz MR, Liu Y, Samimi MK, Sun S, Rangan S, Rappaport TS, Erkip E (2014) Millimeter wave channel modeling and cellular capacity evaluation. IEEE J Sel Areas Commun 32(6):1164–1179
2. Akyildiz IF, Gutierrez-Estevez DM, Balakrishnan R, Chavarria-Reyes E (2014) LTE-advanced and the evolution to beyond 4G (B4G) systems. Phys Commun 10:31–60
3. Alqudah YA, Tahat A (2011) Path loss and propagation models at 3.5 GHz using deployed WiMAX network. In: The international conference on information networking 2011 (ICOIN2011). IEEE, New York, pp 301–305
4. Banchs A, Breitbach M, Costa X, Doetsch U, Redana S, Sartori C, Schotten H (2015) A novel radio multiservice adaptive network architecture for 5G networks. In: 2015 IEEE 81st Vehicular technology conference (VTC Spring). IEEE, New York, pp 1–5
5. Bhushan N, Li J, Malladi D, Gilmore R, Brenner D, Damnjanovic A, Sukhavasi RT, Patel C, Geirhofer S (2014) Network densification: the dominant theme for wireless evolution into 5G. IEEE Commun Maga 52(2):82–89
6. Holma H, Toskala A (2009) LTE for UMTS: OFDMA and SC-FDMA based radio access. Wiley
7. Jha A, Saha D (2017) Why is 700 MHz band a good proposition for provisioning pan-India 4G LTE services? In: Proceedings of 9th international conference on communication systems & networks, pp 1–8
8. Jha A, Saha D (2019) Coverage and capacity dynamics in 4G-LTE deployment in India. In: 2019 International conference on electronics, information, and communication (ICEIC). IEEE, New York, pp 1–8
9. Jha SK, Rokaya R, Bhagat A, Khan AR, Aryal L (2017) LTE network: coverage and capacity planning—4G cellular network planning around Banepa. In: 2017 International conference on networking and network applications (NaNA), pp 180–185. https://doi.org/10.1109/NaNA.2017.23
10. Johansson K, Furuskar A (2005) Cost efficient capacity expansion strategies using multi-access networks. In: 2005 IEEE 61st vehicular technology conference, vol 5. IEEE, New York, pp 2989–2993
11. Johansson K, Zander J, Furuskar A (2007) Modelling the cost of heterogeneous wireless access networks. Int J Mob Network Des Innov 2(1):58–66
12. Mahajan V, Muller E, Wind Y (2000) New-product diffusion models, vol 11. Springer Science & Business Media

13. Markendahl J (2011) Mobile network operators and cooperation: a tele-economic study of infrastructure sharing and mobile payment services. PhD thesis, KTH
14. Markendahl J, Mäkitalo Ö (2010) A comparative study of deployment options, capacity and cost structure for macrocellular and femtocell networks. In: 2010 IEEE 21st international symposium on personal, indoor and mobile radio communications workshops. IEEE, New York, pp 145–150
15. Nikolikj V, Janevski T (2014) A cost modeling of high-capacity LTE-advanced and IEEE 802.11 AC based heterogeneous networks, deployed in the 700 MHz, 2.6 GHz and 5 GHz bands. In: MoWNet, pp 49–56
16. Oughton EJ, Frias Z (2018) The cost, coverage and rollout implications of 5G infrastructure in Britain. Telecommun Policy 42(8):636–652
17. Oughton EJ, Frias Z, van der Gaast S, van der Berg R (2019) Assessing the capacity, coverage and cost of 5G infrastructure strategies: analysis of The Netherlands. Telematics Inform 37:50–69
18. TRAI (2004) The Indian telecom services performance indicators July-Sept'04
19. Wisely D, Wang N, Tafazolli R (2018) Capacity and costs for 5G networks in dense urban areas. IET Commun 12(19):2502–2510

Q-TOMEC: Q-Learning-Based Task Offloading in Mobile Edge Computing

Fatema Vhora, Jay Gandhi, and Ankita Gandhi

Abstract Mobile edge computing (MEC) is a unique approach that facilitates compute-intensive applications on mobile devices. MEC emerges as a promising paradigm to provide computation capabilities to edge servers that are within the closest proximity to the user. Task offloading is an essential issue when multiple devices with several applications are available in the wireless cellular network. In this paper, the authors consider the scenario where multiple devices offload tasks to two access points, including the cloud server and MECServer. The allocations of resources are in the form of virtual machines and different wireless channels for various servers. The authors formulate the problem of minimizing response time by offloading the tasks to cloud and MEC servers for parallel task processing. The proposed algorithm Q-learning-based task offloading in mobile edge computing (Q-TOMEC) used Q-learning-based optimization system to efficiently manage the resource allocation in the MEC scenario by task offloading. The proposed approach is comprised of the "no offloading" scheme where either cloud or MEC makes task execution. The simulation result shows that Q-TOMEC provides a significant improvement in terms of response time, task execution ratio, and bandwidth utilization.

Keywords Mobile edge computing · Task offloading · Resource allocation · Q-learning

F. Vhora (✉)
Department of Information Technology, Dharmsinh Desai University, Nadiad, Gujarat, India
e-mail: fatema.vhora18@gmail.com

J. Gandhi · A. Gandhi
Computer Engineering Department, Parul Institute of Engineering & Technology, Parul University, Waghodia, Gujarat, India
e-mail: jay.gandhi2881@paruluniversity.ac.in; ankita.gandhi@paruluniversity.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_4

1 Introduction

The exponential growth and use of intelligent devices covering smartphones, laptops, and wearable devices increase computationally intensive applications such as augmented reality, image/video processing, face recognition, and interactive gaming. The users endure the mentioned applications in their devices, but they lack battery and computation capacity. So, the task offloading concept is taken into consideration to solve the problem. It refers to moving the computation-intensive tasks into external platforms like a cloud or MECServer [1, 2]. If tasks are offloaded to the cloud, it causes notable transmission delays in networks. Thus, offloading tasks into MECServer provide a lower latency rate, because it is closer to the user's proximity [3–5]. In the existing approach for multiple users, researchers have considered either a single access point for offloading: cloud or local edge server or multiple edge servers which execute tasks simultaneously [6, 7]. The delay in multi-edge servers happens because collecting responses from different servers leads to delay. In single access points, computation and storage capacities are also limited; however, it is higher than user devices and that causes execution delay in heavy traffic [8, 9]. In the proposed approach, authors present offloading decision systems for multiple users, including many independent tasks.

The main contributions of the paper are summarized as follows. (a) To reduce the load of the local device, we formulate the problem of task offloading in which computation-intensive tasks are shifted to the external server, including cloud and MECServer. By calculating tasks size and availability of data in the device, the decision is taken for task offloading by applying the CUCKOO model. (b) In today's era, Internet speed, also considered network response to the user, is lacking. To overcome mentioned issue, our research proposed an approach of parallel tasks processing in two servers simultaneously. The tasks are considered independent tasks in a network that means one task's input is not dependent on the other task's output. (c) The parallel task processing using the Q-learning algorithm is developed to reduce network load, and more resources are available for execution. So, the network executes tasks rapidly and gives fast responses to the user. (d) The proposed approach uses LTE and WLAN channels. Due to dual-channel, the bandwidth of the network also increased, which is helpful for fast virtual machine allocation and task execution for minimizes response time and increases task success ratio.

2 Related Work

To carry out the task offloading process, the prior study assumes that a single access point is enough for the computation of tasks with a low-latency rate [10]. In [10], the authors develop dynamic offloading decision-making for a single access point environment using the supervised learning algorithm. In the advancement of the previous work, multi-server scenarios have been established that consider different numbers of users, i.e., one or more. The authors of [13, 14] focus on latency rate decrement by providing CPU data and task information to linear or semi-linear regression approach. In [15], the authors find the optimal solution for offloading decisions by applying the Markov approximation approach for multi-server, single-user. Authors [7, 11] focused on offloading solutions with a task scheduling approach. Due to the availability of multiple tasks in the network, that utilizes time division multiple access (TDMA) and frequency division multiple access (FDMA). The authors of [12] use the cloud as a single access point with scheduling algorithms. The realistic approach, multi-user and multi-server, is beneficial for users that deliver fast responses and less latency. The principal agenda for such a scenario is rapid response time, for that in [16, 17], authors have proposed an approach that provides input task data and the device's current location to different offloading algorithms. In [18, 20], the authors used the current bandwidth of the channel and length of a time slot as input for reinforcement learning. For edge-assisted servers as their access point, authors in [21] used channel gain of the current network. In the distributed approach [20], a proposed algorithm works on CPU cycles and data size as an input. The several algorithms applied for offloading decision, and task execution includes task information, channel bandwidth with channel gain, CPU cycle, and CPU frequency as the input for the ease of training and testing. The N numbers of layers have been provided for getting optimal solutions in DNN [7, 11]. The DQN [22] used an optimal policy for offloading multiple tasks into multiple servers. Reinforcement learning [7, 23] has adaptive capabilities that the current network condition/environment can learn. The authors from research papers suggest that to carry out task offloading research, one needs to consider (a) offloading scenario and (b) algorithms. In Table 1, several offloading scenarios are presented that can be applied in the research, and their advantages and limitations. The comparison between different scenarios shows that the first two scenarios cannot be used in real life because; in a network, it is least possible that for one user private server or multiple servers provided, practically, it is resource wastage. The remaining two scenarios can be applied depending on the requirements. Table 2 presents the machine learning-based algorithms used for MEC. The analysis of different algorithms gives the objectives along with advantages and limitations. Comparison of different algorithms indicates that most of the algorithms are time-consuming because of the large amount of data feed for training purposes. DQN is model-free learning, but it requires much time for training. Markov approximation gives the most reliable results for optimization but is used only for single devices. Generally, the purposes of each algorithm are to find optimal solutions for offloading decisions that require some necessary information.

To sum up, for this research, the authors have used multi-user and two-server scenarios. The two servers, namely cloud and MEC server, are used for task execution. The parallel execution of the proposed approach takes less time to execute a task and provides a fast response to users. The machine learning-based Q-learning has been chosen for the algorithm because it takes less time in learning and rapidly provides the offloading decision. The research for task offloading is related to the mobile environment, so Q-learning easily interacts with the environment because of its model-free learning [22].

Table 1 Comparative analysis of various offloading scenario of MEC

Offloading scenario	Necessary information	Objectives	Advantages	Limitations
Single-user, single-server [10]	CPU cycles	To develop dynamic offloading framework for mobile users	No congestion, no resource limitation, no extra energy consumption	Mostly used in idle condition
Multi-user, single-server [7, 11, 12]	Link information, channel gain, workload	Offloading with task scheduling	No duplication of data to multiple server	Resource management, congestion control, more energy consumption
Single-user, multi-server [13–15]	CPU frequency, input task data	To find optimal solution with minimum latency	No congestion, no resource limitation	Mostly used in idle condition
Multi-user, multi-server [5, 16–20]	Current location of device, input task data	To minimize task execution latency and energy	Easy offloading, quick response time, more availability of resources	Combining responses from different server is challenging

3 System Model

In this section, the authors presented a system architecture that is used for parallel task processing. After that, the system's flow is comprised, which shows the decision-making process for the task offloading. Subsequently, the proposed Q-TOMEC algorithm is presented along with methodologies applied in the proposed system.

3.1 System Architecture

Figure 1 comprises multiple elements that include n number of wireless devices, n number of tasks for each device, single cloud server, and single MECServer.

The authors assume that there are two kinds of mobile devices, busy mobile devices that have some applications running, and other is idle mobile devices without any applications. If n numbers of busy devices are in the network, each device has n number of independent tasks. For the parallel task execution, the computation-intensive tasks offload to either cloud or MECServer for the execution. Both the servers have multiple virtual machines (VM) used to execute offloaded tasks. Due to the parallel execution of n tasks, the user gets a fast response. For the cloud, the LTE channel is used, and for the MECServer, WLAN channel is used.

Table 2 Comparison of machine learning-based algorithms used in MEC

Algorithm/techniques	Necessary information	Objectives	Advantages	Limitations
Deep neural network (DNN) [11]	A large amount of task data, latency rate, channel bandwidth	To generate optimal decisions within a fraction of second	N number of layers and input can be taken, works well with more inputs	Time-consuming, expensive, variable influences are there
Deep reinforcement learning [7, 18]	A large amount of task data, latency rate, channel bandwidth	To maximize CPU cycle and reduce energy	Learn from experience, useful for environment scenario	Time-consuming
Distributed deep learning [20]	Large amount of task data, latency rate, channel bandwidth	Generate offloading decision	Once the model is trained, it will give near to perfect result	lot of data required, time consuming
Deep Q network (DQN) [22]	State action information, channel bandwidth,	To offload multiple tasks to multiple server	Use optimal policy for Q-table	Variable influences are there
Markov approximation [15]	Input data, required CPU cycles	To find optimal offloading decision	Great for joint optimization for CPU frequency scaling and task assignment	Single mobile device offloading
Linear regression [14]	CPU frequency	Based on prediction generate offloading decision	Best for single-user network	Used for LAN network only

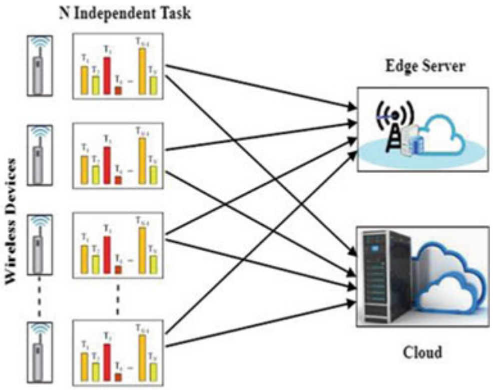


Fig. 1 Proposed system architecture for parallel execution

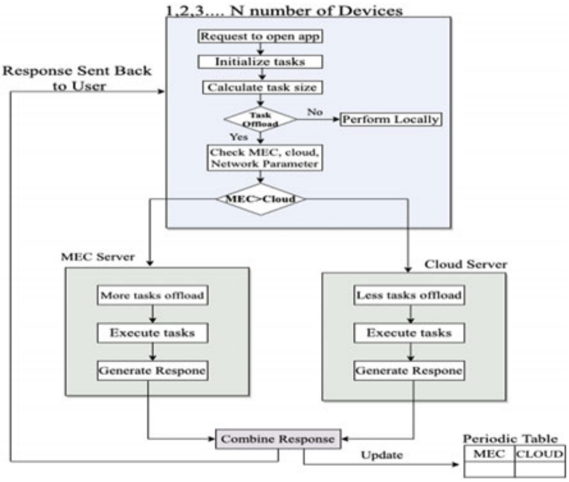


Fig. 2 Flowchart of the Q-TOMEC for offloading decision

3.2 Flowchart for Making the Offloading Decision

Figure 2 shows the detailed flow of the system that starts from the opening application, initialization of tasks, and the input data size of tasks are calculated. First, the decision needs to be taken regarding whether the tasks offload or perform locally using the CUCKOO model [23]. If the tasks are offloaded, then the second decision that needs to be taken is on which server we can offload it depending on network parameters, task parameters, MEC parameters, cloud server parameters, etc. These parameters are used to generate a reward for the server by applying Q-learning [22].

3.3 Proposed Algorithm

In this section, the distributed offloading algorithm based on Q-learning is proposed to minimize response time. In the proposed algorithm Q-TOMEC, channel bandwidth and no. of available mobile devices at a time are given as input for specific simulation time. The main output of the algorithm is the task offloading decision and depends on that update in the reward table. Based on no. of mobile devices, each device has n number of tasks. So in step 1, initialization of N devices and n tasks is made. Each device's n task algorithm generates offloading decisions using the CUCKOO model in steps 2–3. To apply the CUCKOO model, the length of tasks, and availability of data at the local device, these two parameters are considered as shown in steps 4–7.

$$Q(s,a) = ((1-\alpha) * Q(s,,,a)) + \alpha * \left(r(s,,,a) + \gamma \max Q\left(s', a'\right)\right) \qquad (1)$$

Here, α Learning rate, γ-discount factor.

Q(s,a) initial value(initialize randomly)/old value.
r(s,a) reward for each action, max Q(s’,a)-estimate of optimal future value

Algorithm 1: Q-Learning based Task Offloading in Mobile Edge Computing (Q-TOMEC)

```
INPUT:      Simulation time (T),
            No. of devices (N),
            Channel Bandwidth (B)
OUTPUT:     Offloading decision (x),
            No. of completed tasks (tc),
            Avg. service time (st)
            No. of failed tasks (tf),
            No. of failed task due to mobility (tm)
            No.of failed task due to dead device (tdd)
```

```
Initialize N no. of devices with n no. of tasks
    N={1,2,3….N)
    n = {1,2,3….n}
for 1 to N
  |   for 1 to n do
  |     |   Generate offloading decision x= x d,t (CUCKOO Model)
  |     |   If   x=0
  |     |     |   Perform task locally
  |     |   end
  |     |   else
  |     |     |   Find reward r= SXA→Q by solving eq (P1) then
  |     |     |   Offload tasks to MEC and Cloud Server
  |     |   end
  |   End
  |   Combine response from both servers and sent response to user
End
Update periodic table
```

$$X\ d,t\ \text{(COCKOO Model)} \begin{cases} 0 & \text{d=0, t=0 (d=data=0=Locally available=task=0= Locally perform)} \\ 1 & \text{Otherwise} \end{cases}$$

P1: $Q: SXA \rightarrow R$

S=State
A= Action
R=Reward
Q= Q-table value

If the length of tasks is >1000 and data is not available at the local device, then the task offloads to the external platform. If data is available on the device and the length of tasks is <1000, then it performs locally; the length of tasks is calculated while initialization of tasks and stored in the table. Based on the before-mentioned parameters model gives a binary result, i.e., 0 or 1. If the model gives 0 as a result, then the task performs locally; otherwise, task offloads to an external server. Once the decision is taken for all the available tasks on the network, Q-learning is applied to generate a reward for the server. Q-learning is a model-free reinforcement learning used to learn the current network condition and make appropriate decisions. In steps 8-12, using current channel bandwidth, n number of states generated and two actions (1) offloaded to cloud and (2) offloaded to MECServer is defined. At the time of random initialization, a reward is generated for both servers. The state combines WLAN bandwidth, LTE bandwidth, uplink data rate, downlink data rate. The bandwidth decrement policy is applied for a state change. The Q-table is generated with periodic rewards; tasks are allotted based on the maximum reward for a specific state. Based on the successful task execution ratio, the server gets punished or gets a reward by applying Q-learning equation-1, and the appropriate reward is updated. After the successful task execution in the allocated server, in step 13, responses are combined from servers and sent to clients. In step 15, the periodic table is updated after sending a response to the client. The Q-TOMEC algorithm generates results in terms of no. of completed tasks (tc), no. of failed tasks (tf), the response time (st), no. of failed task due to mobility (tm), no. of failed task due to dead device (tdd) is a model-free reinforcement learning used to learn the current network condition and take appropriate decisions.

4 Simulation Results

For the simulation result, the authors used a “PureEdgeSim” [24] simulator to measure the performance of the proposed Q-TOMEC algorithm.

The architecture of PureEdgeSim depends on CloudSim plus [25]. It provides many inbuilt features like edge configuration, fog server, and cloud configuration. It contains layered architecture, which is beneficial for communication between components of PureEdgeSim. It has an extensible library from cloud resources like data center and host to services such as VM allocation policies, CPU scheduler [24]. In a simulator, the authors have configured the cloud server and MEC server.

The various computation-intensive applications, including augmented reality, E-health, Pokémon Go, Google lens, are utilized in the network. The unique id is provided to the applications to differentiate the different applications running on the device. Table 3 shows the network parameters applied in the system. N = 500 devices are given as input in the simulation, and a random number of tasks are generated. The offloading decision is made using the data stored in the table in the form of 0 and 1, where 0 = locally perform, 1 = offloaded to a server. After that, Q-learning is applied to the tasks which are allotted to the server for each state. The state is generated by the composite components, including LTE bandwidth, WLAN bandwidth, uplink data rate, downlink data rate. The parameters used for Q-learning are defined in Table 4,

Table 3 Network parameters

Parameter name	Description
Input data size of task	Input task size
Output data size of task	Output task size
LAN bandwidth	Capacity of local area network (LAN) channel to transmit maximum amount of data from one point to another point via Internet
WLAN bandwidth	Capacity of wireless local area network (WLAN) channel to transmit maximum amount of data from one point to another point via Internet
LTE bandwidth	Capacity of long-term evolution (LTE) channel to transmit maximum amount of data from one point to another point via Internet
Mobile storage	Capacity of your mobile to process data/store data
Uplink data rate	Time required (data rate) to transmit data from mobile device to base station
Downlink data rate	Time required (data rate) to transmit data from base station to mobile device

in which parameters, their description, and elements used for the proposed system are described. Finally, the server receives the reward of + 1, which has successfully executed tasks ratio above 85% from allotted tasks, and the server is punished with a −1 reward which has below 85% ratio. By applying Q-learning equation-1, the reward is updated in Q-table for future reference. The data stored in the Q-table is shown in Table 5. Q-table is a combination of states and actions, and based on that, the reward is generated.

Each server has 8 virtual machines (VM). The tasks are divided over VM for execution based on maximum reward policy, and response is produced from the server. Responses from servers are combined and sent to the client. After task execution, depending on selected network architecture, i.e., only cloud, only MEC, or Q-TOMEC, response time is calculated along with no. of successfully executed tasks, no. of failed tasks, and bandwidth utilization is calculated. The comparison ratio is shown in Table 6.

Table 6 shows the comparison of response time for different network architecture, including only cloud architecture that is a single access point cloud server with eight virtual machines. The only MEC architecture has a single access point, MECServer, with eight virtual machines but less storage than the cloud. The proposed approach Q-TOMEC has two access points, cloud as well as MECServer with 16 virtual machines. As shown in Table 6, it is observed that the proposed approach has minimum response time, highest successfully executed tasks ratio, lowest failed task ratio due to mobility and dead device.

Figure 3 represents the graph that calculates response time for cloud, MEC, and Q-TOMEC. The graph shows that for up to 100 devices, there is no significant difference in response time. However, the task execution is delayed when the number of devices increases due to the lack of virtual machines in the only cloud, only MECServer

Table 4 Q-learning parameters

Parameter	Description	In proposed system
Agent	Hypothetical entity which perform actions in an environment to gain some reward	Mobile device
Environment	A scenario the agent has to face	Multi-user, multi-tasks, two-server (Fig. 1)
State	Current situation return by environment	Composite state S = {LTE bandwidth, WLAN bandwidth, Uplink data rate, Downlink data rate}
Action	All possible moves that agent can take	Offload to MEC, offload to cloud
Reward	An immediate return sent back from environment to evaluate last action by agent	Maximum reward of server get majority tasks for execution
Policy	The strategy that agent employs to determine next action based on current state	Bandwidth decrement
Learning rate	Determines the extent of newly acquired information can inflate	0.1 (0 < LR < 1)
Discount factor	Determines importance of future reward	0.9 (0 < DF < 1)

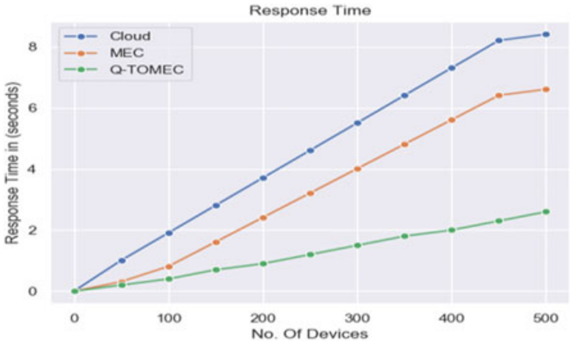
Table 5 Example: Q-table for with its reward values

Action			
States		A1 Offload to MEC	A2 Offload to Cloud
	S0	1.2	0.1
	S1	1.5	0.8
	S2	0.5	1.7
	S3	1.9	0.2
	…	…	…
	Sn	1.4	0.8

Table 6 Comparison between existing and proposed approach for N = 500 devices t = 439,811 offloaded tasks

Scenario	Time in seconds	Successfully executed tasks	Task failure	Task failed due to mobility	Task failed due to dead device
Only cloud	8.409	394,402	45,409	27,919	17,490
Only MEC	6.611	404,578	35,233	22,601	12,632
Q-TOMEC	2.601	435,412	4399	3379	1020

Fig. 3 Response time



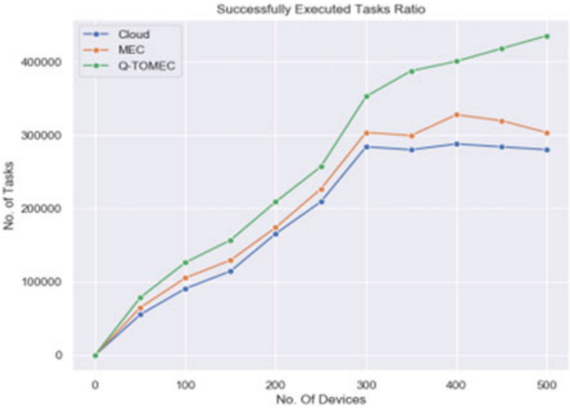
architecture. The proposed architecture has more virtual machines than the other two that provide minimum response time in case of more no. of devices.

Figure 4 represents the graph that shows the tasks ratio of successful task execution on cloud, MEC, and Q-TOMEC. The simulation result shows that when numbers of devices are less in the network, i.e., up to 300 devices, there is not much difference in the ratio of successfully executed tasks. After 300 devices in a network, the success ratio is decreased for cloud and MEC, but for Q-TOMEC, success ratio is increased.

Figure 5 shows the task failure ratio for cloud, MEC, and Q-TOMEC. The result shows that the failure ratio is increasing for cloud and MEC architecture as the number of devices increases along with the number of tasks. The main reason for the low-failure ratio in Q-TOMEC is two access points with almost double virtual machine capacity and parallel tasks processing of tasks.

Figure 6 shows the task failure ratio that is caused due to mobility (i.e., devices are changing location or leaving the territory of network) for cloud, MEC, and Q-TOMEC. It is observed that the cloud server has the highest task failure ratio, while Q-TOMEC has the lowest task failure ratio. The reason behind low-failure tasks due

Fig. 4 Successfully executed tasks ratio



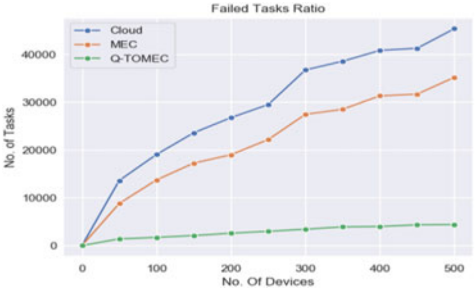


Fig. 5 Failed tasks ratio

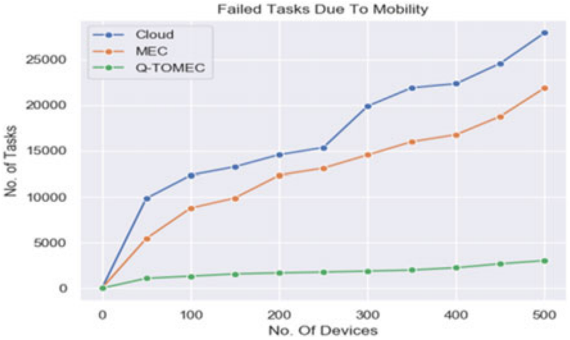


Fig. 6 Failed tasks due to mobility

to mobility is, while roaming around the network if the connection is lost with one server, there is always a second server in the network to connect with.

Figure 7 shows the task failure ratio that is caused due to device dead (i.e., the device is running out of battery and switched off) for cloud, MEC, and Q-TOMEC. It is observed from the graph that the MEC server has a lower failure ratio compared to the cloud and a higher failure ratio compared to Q-TOMEC. At the same time, Q-TOMEC has the lowest failure ratio, and cloud has the highest failure ratio. Due to the fast response time, the time taken to generate a response for the proposed approach is minimal, saving the device's energy and low-task failure ratio. Figure 8 shows the bandwidth utilization for three different approaches. The cloud server has 90 Mbps bandwidth, and MEC server has 150 Mbps bandwidth, while Q-TOMEC has advantages of both channels' bandwidth (i.e., 150 + 90 = 240 Mbps). So, when the network reaches 500 devices, then cloud bandwidth decreases to 25, MEC reaches up to 50 while Q-TOMEC still manages to 150 Mbps which gives a fast response to the user.

Fig. 7 Failed tasks due to dead device

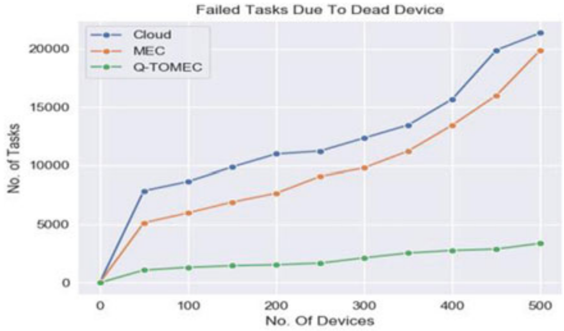
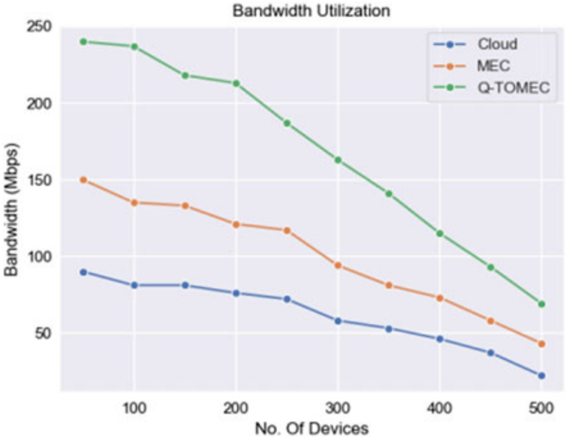


Fig. 8 Bandwidth utilization



5 Conclusion and Future Work

In this paper, author has studied a Q-learning-based optimization framework for task offloading and resource allocation in wireless cellular networks using MEC. The proposed Q-TOMEC algorithm formulates to take offloading decisions that minimize the response time among users and network using parallel task processing between cloud and MEC servers. Effective resource allocation minimizes response time, increases no. of successfully executed tasks, decreases no. of failed tasks, and maximizes the bandwidth utilization in the network. The proposed algorithm performs task offloading using cloud and MEC servers with parallel execution to achieve the best performance. The adaptive Q-learning algorithm helps to obtain the optimal offloading decision. Finally, the performance evaluation of the proposed approach is presented in comparison with the "no offloading" approach. The simulation results demonstrate that the Q-TOMEC performs better in terms of minimum response time, maximum successfully executed tasks ratio, minimize failed tasks ratio, and highest bandwidth utilization. Future work will include more than one MECServer for multi-user multi-system scenarios to enhance performance.

References

1. Kumar K, Liu J, Lu Y-H, Bhargava B (2013) A survey of computation offloading for mobile systems. Mobile Netw Appl 18(1):129–140
2. Mai T, Yao H, Guo S, Liu Y (2021) In-network computing powered mobile edge: Toward high performance industrial iot. IEEE Network 35(1):289–295
3. Yan J, Bi S, Zhang Y-JA (2018) Optimal offloading and resource allocation in mobile-edge computing with inter-user task dependency. IEEE Global Commun Conf (GLOBECOM) 2018:1–8
4. Bhattacharya P, Tanwar S, Shah R, Ladha A (2020) Mobile edge computing-enabled blockchain framework—a survey. In: Singh P, Kar A, Singh Y, Kolekar M, Tanwar S (eds) Proceedings of ICRIC 2019. Lecture notes in electrical engineering, vol 597. Springer.
5. Zeng J, Sun J, Wu B, Su X (2020) Mobile edge communications, computing, and caching (Mec3) technology in the maritime communication network. China Commun 17(5):223–234. https://doi.org/10.23919/JCC.2020.05.017
6. Wolski R, Gurun S, Krintz C, Nurmi D (2008) Using bandwidth data to make computation offloading decisions. In: 2008 IEEE international symposium on parallel and distributed processing, pp 1–8. https://doi.org/10.1109/IPDPS.2008.4536215
7. Huang L, Bi S, Zhang Y-JA (2020) Deep reinforcement learning for online computation offloading in wireless powered mobile-edge computing networks. IEEE Trans Mob Comput 19(11):2581–2593
8. Fan Y, Zhai L, Wang H (2019) Cost-efficient dependent task offloading for multiusers. IEEE Access 7:115843–115856
9. Vhora F, Gandhi J (2020) A comprehensive survey on mobile edge computing: challenges, tools, applications. In: 2020 fourth international conference on computing methodologies and communication (ICCMC), pp 49–55
10. Yu S, Wang X, Langar R (2017) Computation offloading for mobile edge computing: a deep learning approach. In: 2017 IEEE 28th annual international symposium on personal, indoor, and mobile radio communications (PIMRC), pp 1–6
11. You C, Huang K, Chae H, Kim B-H (2017) Energy-efficient resource allocation for mobile-edge computation offloading. IEEE Trans Wireless Commun 16(3):1397–1411. https://doi.org/10.1109/TWC.2016.2633522
12. Zhou B, Dastjerdi AV, Calheiros RN, Buyya R (2018) An online algorithm for task offloading in heterogeneous mobile clouds. ACM Trans Internet Technol 18(2):1–25. https://doi.org/10.1145/3122981
13. Thinh TQ, Tang J, La QD, Quek TQS (2017) Offloading in mobile edge computing: task allocation and computational frequency scaling. IEEE Trans Commun, pp 1–1. https://doi.org/10.1109/TCOMM.2017.2699660
14. Hu YC et al (2015) Mobile edge computing—a key technology towards 5G[ETSI white paper]11(11):1–16
15. Zhou W, Fang W, Li Y, Yuan B, Li Y, Wang T (2019) Markov approximation for task offloading and computation scaling in mobile edge computing. Mob Inf Syst 2019:1–12. https://doi.org/10.1155/2019/8172698
16. Chen X, Chen S, Ma Y, Liu B, Zhang Y, Huang G (2019) An adaptive offloading framework for Android applications in mobile edge computing. Sci China Inf Sci 62(8):82102. https://doi.org/10.1007/s11432-018-9749-8
17. Tran TX, Chan K, Pompili D (2019) Costa: Cost-aware service caching and task offloading assignment in mobile-edge computing. In: 2019 16th annual IEEE international conference on sensing, communication, and networking (SECON), pp 1–9

18. Dinh TQ, La QD, Quek TQS, Shin H (2018) Learning for computation offloading in mobile edge computing. IEEE Trans Commun 66(12):6353–6367
19. Chen X, Jiao L, Li W, Fu X (2016) Efficient multi-user computation offloading for mobile-edge cloud computing. IEEE/ACM Trans Networking 24(5):2795–2808
20. Huang L, Feng X, Zhang L, Qian L, Wu Y (2019) Multi-server multi-user multi-task computation offloading for mobile edge computing networks. Sensors 19(6):1946
21. Sheng J, Hu J, Teng X, Wang B, Pan X (2019) Computation offloading strategy in mobile edge computing. Information 10(6):191. https://doi.org/10.3390/info10060191
22. Huang L, Feng X, Zhang C, Qian L, Wu Y (2019) Deep reinforcement learning-based joint task offloading and bandwidth allocation for multi-user mobile edge computing. Digital Commun Networks 5(1):10–17
23. Kemp R, Palmer N, Kielmann T, Bal H (2012) Cuckoo: a computation offloading framework for smartphones. In: Gris M, Yang G (eds) Mobile computing, applications, and services. MobiCASE 2010. Lecture notes of the institute for computer sciences, social informatics and telecommunications engineering, vol 76. Springer, Berlin, Heidelberg
24. Mechalikh C, Taktak H, Moussa F (2019) Pureedgesim: a simulation toolkit for performance evaluation of cloud, fog, and pure edge computing environments. In: 2019 international conference on high performance computing & simulation (HPCS), pp 700–707
25. Filho MCS, Oliveira RL, Monteiro CC, Inacio PRM, Freire MM (2017) CloudSim plus: a cloud computing simulation framework pursuing software engineering principles for improved modularity, extensibility and correctness. In: 2017 IFIP/IEEE symposium on integrated network and service management (IM), pp 400–406

Mobility Management in Heterogeneous Network Using Systematic Hierarchy Process and Seagull Optimization Algorithm

S. S. Sambare and M. U. Kharat

Abstract Mobile Nodes (MN) can generally be interfaced with different wireless networks, changing the quality of service for empowering the delivery of different classes of services. Multi-Attribute Decision Making (MADM) method provides a well-organized method intended for maintaining competitive networks in addition to selecting the optimal one as indicated by the network parameters. In literature, some of the methods are reviewed for enabling the proper handover process. However, these methods are not providing the efficient handover process in heterogeneous networks. Hence, in this paper, Systematic Hierarchy Process (SHP) and Seagull Optimization Algorithm (SOA) are proposed to enable proper handover in heterogeneous networks. The proposed methodology is utilized to enable the efficient and quality communication in heterogeneous networks. This method is utilized to optimize the network parameters by ranking network based on quality and reducing unwanted handovers. The SHP process is utilized to select and ranking the network connections based on their quality parameters. To empower the performance of SHP, the SOA is utilized to select optimal weighting parameters. This technique is implemented in NS2 in addition performances are assessed by performance metrics like latency, network lifetime, energy consumption, throughput, delivery ratio, drop and delay. The proposed methodology is compared with the conventional methods like Genetic Algorithm (GA) and Particle Swarm Optimization (PSO). The proposed method has achieved the performance metrics such as energy consumption: 0.14 J, latency: 0.012 s, network lifetime: 62 s, throughput: 1452 kbps, and delivery ratio: 0.95 s.

Keywords Handover · Mobility management · Seagull optimization algorithm · Systematic hierarchy process · Communication · Quality measurements

S. S. Sambare (✉)
PCCOE, Pune, India
e-mail: santosh.sambare@pccoepune.org

M. U. Kharat
METIoE, Nashik, India

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_5

1 Introduction

In new generation of communication systems, the mobile nodes (MN) are utilized in a heterogeneous network for different services such as multiple interfaces, real services, and non-real services. The recent enhancement in connections of wireless devices brings different issues and challenges to make a generic and efficient communication design intended for upcoming systems [1]. The handover technology improvements are essential to empower the consistency of heterogeneous networks [2, 3]. Due to the incompatibility of the heterogeneous networks, it is increased and network covered the complete world through different applications like smart societies, cities, and smart homes with consideration of IoT [4]. The accessibility of different resources, services, devices, and objects is now possible with the development of cloud computing, Cyber-Physical Systems (CPS), Machine to Machine Communication (M2M), and IoT [5]. In recent years, mobile phone applications and technologies are utilized widely.

The modern mobile phones are connected with different interfaces to access various networks such as ZigBee, Bluetooth, WiMAX, and WIFI [6]. To empower the operation of the mobile phone technologies, efficient methods can be developed toward handover an MN in the heterogeneous network for empowering the performance of the system before failure occurred in the system [7].

Moreover, the researchers are computed higher energy efficient nodes among the complete nodes which is selected to best node for initiating the interface in the network. The MN has the ability to switch among the interfaces by following a specific profile for every interface. The MN is connected the interface automatically [3]. The automatically connected with the network may be provided the poor connection. Hence, selection of best network is essential for enabling quality networks in mobile technologies [8]. So, the optimal selection of networks concept is an important factor in mobile technologies such as M2M, CPS, and IoT. The network selection is considered in the handover management which enables the efficient communication in the mobile networks [9].

The handover process has the two types such as hard handover and soft handover. The soft handover can be divided into three types such as diagonal, vertical, and horizontal handover [10]. The hard handover is defined as break the old connection and enable the new connection in the mobile nodes. So, the traffic flow is initially shielded with the old system in addition it is readdressed to new assembly. In the mobile technologies, hard handover is utilized to enable the efficient communication [11]. The literature contains different methods for the handover triggering but, it does not provide efficient handover triggering technique. Additionally, different parameters are considered to enabling the handover such as Jitter, Bandwidth, Signal to Interference, and Noise Ratio (SINR) in addition Received Signal Strength (RSS) [12, 13]. With the consideration of signal strength, the handover process is enabled in the mobile nodes. Moreover, different challenges are considered utilizing a single value aimed at enabling handover in mobile nodes. Recently, Artificial Intelligence (AI) is utilized to enabling efficient handover processes in the mobile networks such as Artificial Bee Colony (ABC) algorithm, Particle Swarm Optimization (PSO), and Genetic Algorithm (GA) [14].

Main contribution and organization of the work

- In this paper, SHP and SOA is proposed to enable proper handover in heterogeneous network.
- The efficient and quality communication is enabled by using proposed SHP and SOA techniques. To reducing unwanted handover and enabling efficient handover process, this proposed methodology is utilized by ranking networks based on their network parameters.
- The SHP process is utilized to select and ranking the network connections based on their quality measurements. To select the optimal weighting parameters of the SHP, SOA are utilized which enhances the performance of the handover process.
- This technique is implemented in NS2 in addition performances are assessed by performance metrics like latency, network lifetime, energy consumption, throughput, delivery ratio, drop and delay. The proposed methodology is compared with the conventional methods like GA and PSO.

The remaining section of the article is pre-arranged as follows, Sect. 2 given the detailed review of the handover-related works. Section 3 provides the brief information of the proposed methodology. Section 4 provides the outcome analysis of the projected methodology. At last, the paper's conclusion is given in Sect. 5.

2 Related Works

Various handover techniques in heterogeneous mobile networks are developed by researchers. Few research works are analyzed in this section.

Khan et al. [15] have presented generic vertical handover management method for enabling efficient communication in heterogeneous networks. This proposed methodology was scheduled with two phases. In initial phase, the mobile node dynamically checks the required data with the consideration of mobile nodes. The data required was computed based on threshold value. The second phase, the network selection was achieved by considering various attributes like end-to-end delay, jitter, bit error rate, and packet loss. To select the optimal network with the less time in addition handover delay, the Artificial Bee Colony (ABC) was utilized.

Beshley et al. [16] have presented a self-optimizing method for load balancing in heterogeneous networks. The structure diagram was developed for wireless heterogeneous networks which were utilized to enhance the efficiency of their functioning. This design was utilized aimed at learning the functioning procedure of a heterogeneous system. This developed design was utilized of big data assessment to validate monitoring of data transmission, statistical output of vertical handover initiation, and analysis of responsibilities created by network users in the mobile communication infrastructure.

Naresh et al. [17] have presented a multi-objective emperor penguin handover optimization (MOEPHO) aimed at optimal handover in communication networks. In the research work, the overall network variables were considered to the vertical handover optimization. So, the efficient handover with less energy consumption and delay has been achieved by using this method. The optimized result was related with the parameter selection to reduce the handover failure rate. Additionally, the availability of different parameter optimization processes was analyzed and discussed in terms of accuracy, latency, and failure rate.

Alhammadi et al. [18] have presented a velocity-based self-optimization algorithm to manage the regulator parameter in 4G/5G systems. The presented technique was utilized the speed and received power to change the time and handover boundary during the user mobility in the network. The mobility management of the self-organizing network was achieved by proposed methodology. The mobility management has been attained by considering the self-optimization algorithm.

Emam et al. [19] have presented cross-layer-based vertical handover calculations based on signaling overhead and signaling. The utilized mobility protocol manages multiple physical paths over heterogeneous networks. The design consisted primarily of three sections, (i) gathering network circumstances, (ii) moving among objects, and (iii) multidisciplinary calculation depending on network circumstances. The cross-layer plot predicts transferable accessibility to the sender. Mathematical results contain improvements brought about through the ability of Indian Standard (IS) to support dynamic information. The proposed architecture can be deployed on any internet protocol layer such as network layer, transport layer, and application layer. Validation results demonstrate the achieved improvements over the reference schemes.

Duong et al. [25] have presented a stochastic geometric analysis method for enabling proper handover and small cell networks to manage the various topologies.

Zaheeruddin et al. [26] have presented a new optimized vertical handoff algorithm for the seamless connectivity of the users to the wireless heterogeneous networks. The proposed algorithm provides better connectivity to the networks with reduced number of redundant handoffs. In the proposed work, all the simulations are done in NS3 simulation network by taking into consideration various parameters which are necessary for taking handoff decisions. Here, using the convex optimization technique with Gradient Descent algorithm for taking optimized Vertical Handoff Decision. The results obtained were then compared with the existing handoff algorithms and it was concluded that the proposed handoff algorithm outperforms the existing ones.

3 Proposed System Model

In recent years, the usage of smartphone technologies is increased for different kinds of applications such as high and low-frequency applications. This kind of application needs a specific QoS level to connect with the MN devices. The proposed system is considered different performance parameters for connecting the MN in the heterogeneous network with the consideration of different features. These features are call drop, delay, delivery ratio, energy, latency, throughput, and network lifetime. Hence, whenever a mobile network is changing from one system to another system, these parameters are considered in the proposed methodology. The mobile network can be given best results when connected with best network connections. Normally, the heterogeneous mobile network contains different handover paths for enabling communication. The efficient handover is selected with the help of proposed methodology with the consideration quality of network. After that, mobile network is connected with selected network connection [20] (Fig. 1).

The selection of network connections is considered as decision-making problem. The decision-making problem is solved by the proposed approach. The proposed approach is consisting of SHP and SOA algorithms. The SHP is utilized to select optimal parameters in mobile networks. In the SHP, the weight parameters are

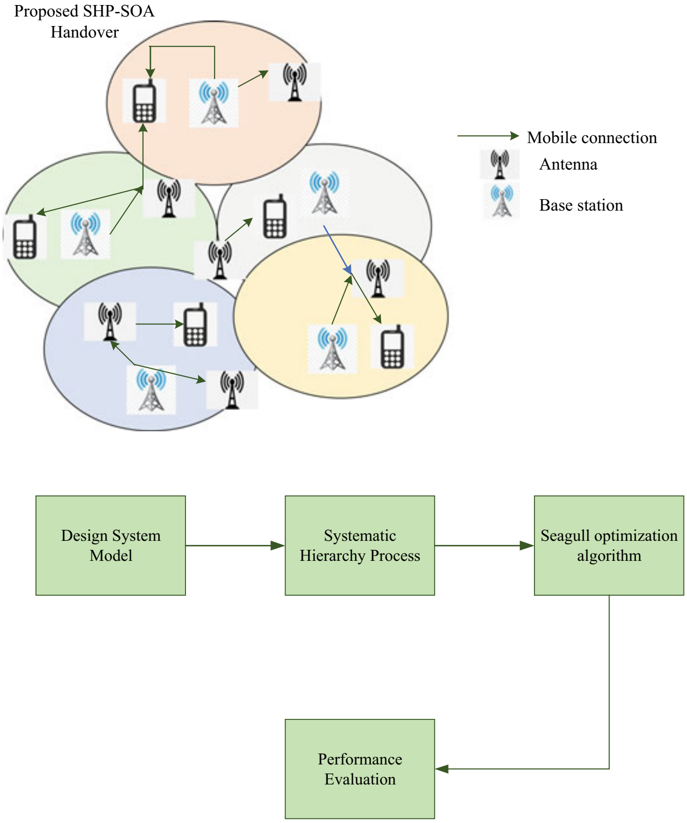

Proposed SHP-SOA Handover
Mobile connection
Antenna
Base station
Design System Model
Systematic Hierarchy Process
Seagull optimization algorithm
Performance Evaluation

Fig. 1 Proposed system architecture [20]

selected with the help of SOA algorithm. The detailed analysis of the projected SHP and SOA is explained in the underneath unit.

Step 1: **Design System Model**
Step 2: **Systematic Hierarchy process**
Step 3: **Seagull Optimization Algorithm**
Step 4: **Evaluation Process**

Initially, the heterogeneous system model is designed. After that, for ranking purposes, the SHP process is utilized in the heterogeneous network. In the SHP, the optimal weighting parameters is selected with the help of SOA algorithm. The proposed methodology is used to enable the efficient and quality communication in heterogeneous networks. The proposed methodology is utilized to optimize the network parameters based on the ranking process which is utilized to reduce the unwanted handover in heterogeneous networks. The SHP process is utilized to select and ranking the network connections based on their quality measurements. To empower the performance of SHP, the SOA is utilized to select optimal weighting parameters. This technique is implemented in NS2 in addition performances are assessed by presentation metrics like latency, network lifetime, energy consumption, throughput, delivery ratio, call drop and delay. The proposed methodology is compared with the conventional methods like Genetic Algorithm (GA) and Particle Swarm Optimization (PSO).

3.1 Systematic Hierarchy Process

The analytical hierarchy process is a procedure that gives a numerical weight to every decision alternative related to how well that alternative completes the condition set for decision making. The created weights related on significance in every parameter contrasted toward different parameters. This technique tries to compute an experience in addition to pair-wise comparisons for developing a decision matrix [21]. And, SHP methods procedure is presented below,

Stage 1: Problem Formulation

Initially, the problem can be formulated into hierarchy that consists of three levels such as.

(1) alternative solutions are presented in bottom level
(2) subsequent level presented a decision factor
(3) topmost level contains the complete objective.

Stage 2: Formulation of a pair-wise comparison matrix

The next stage is processed toward develop a pair-wise comparison matrix to give a computable valuation of the significance of choice conditions by each other. The level hierarchy contains nine-point scale parameters which is utilized to analyze the qualitative judgments based on their quality parameters. For example, the parameter

Table 1 SHP parameter scale

Importance No.	Definition	Description
2, 4, 6, 8	Middle significance	When cooperation can be required
9	Total significance	The indication considering A(I) over A(J) is of the uppermost conceivable
7	Established significance	A(I) is very strong preferred over A(J)
5	Strong or essential significance	Judgments and experience powerfully strong preferred over A(J)
3	Weak significance	Judgments and experience somewhat strong preferred over A(J)
1	Equivalent significance	A(J) in addition A(I) are similarly significant

A is decisively more significant than B it was considered the scale 7. After that, the parameter B is relatively less significant which was assigned to 1/7. The matrix diagonal value is taken as 1 which reproduces slightly presented attribute significance contrasted with itself. Based on formulation, the comparison matrix is generated. After that, the columns are added in addition every admission can be regularized with consideration of column weight. Finally, normalized comparison matrix of every row can be added in addition, alienated with the total number of parameters in the row of comparison matrix. The final outcomes provide vector values in weight that consist of percent weight of every condition which contrasted to the other ones. The SHP importance factor is presented in Table 1.

Stage 3: Computation of the weights for matrix consistency

The last advance pair intelligence selection created the size of each segment of the network and introduced it into the last column. The final framework is standardized to create a standardized network by creating the boundaries of the total grid. The stability ratio should be 0, 1, in which case the selection structure is classified as stable. The calculation of the most reasonable standardization correlations to standardize the unique position is a policy issue in a number of dynamic issues. To solve this problem SOA is used in the proposed system. A detailed description of the proposed system is introduced in the section below.

3.2 Seagull Optimization Algorithm

Seagulls are living in the sea and its scientific named as Laridae which presented whole planet. The extensive variety of seagulls are presented in dissimilar lengths in addition to masses. Seagulls can eat earthworms, amphibians, reptiles, fish and insects, and so on. Seagulls are intelligent species and its body is covered with white plumage. It attacks the fish by utilizing bread crumbs and generates rain similar sound by utilizing bases that entice earthworms concealed below the crushed. It can drink salt water and fresh water. It utilizes their intelligence to attack and find the food. The greatest essential object around the seagulls can be their attacking and migrating characteristics. To compute the abundant and richest food [22], the seagulls are moved which is defined as migration. During migration process, the seagulls are moved from one location to another location. These characteristics are presented as follows,

- During migration process, seagulls are traveled in a cluster. The early locations of seagulls can be various to evade the crashes among apiece additional.
- The seagulls are moved to the best way for attacking food.
- Based on best fitness seagulls, remaining seagulls are updated their locations.

Seagulls are normally bout birds ended the sea whenever moved from one place to another place. During attacks, it creates the spiral shape movement. The mathematical model of the seagull optimization is presented below section.

3.2.1 Mathematical Model

The mathematical formulation of attacking prey and migration process is presented in this section.

Migration

In the migration process, the seagulls are moved from one place to another place. In the migration process, three conditions are satisfied which are presented as follows,

Collision avoiding:

To omit the collision among neighbors, parameter a is added for computation of position of new search agent.

$$\overrightarrow{c_S} = a \times \overrightarrow{p_S}(X) \tag{1}$$

where, X can be described as current iteration, $\overrightarrow{p_S}$ can be described as current location of search agent, $\overrightarrow{c_S}$ can be described as search agent position which prepares not to strike with remaining search agent and a can be described as search agent movement characteristics in a specified search space.

$$a = F_C - \left(X \times \frac{F_C}{\text{Iteration(MAX)}} \right) \tag{2}$$

where, $X = 0, 1, 2, \ldots, \text{Iteration(MAX)}$, a is considered as linearly varied from F_C to 0, F_C can be described as control frequency which is set to 2.

Movement toward optimal neighbors' direction

Once collisions are avoiding among neighbors, the search agents are moved to the optimal location of neighbor [23],

$$\overrightarrow{M_S} = b \times \left(\overrightarrow{p_{\mathrm{bS}}}(X) - \overrightarrow{p_S}(X)\right) \tag{3}$$

where, $\overrightarrow{p_{\mathrm{bS}}}(X)$ can be described as best fitness search agent, $\overrightarrow{p_S}(X)$ can be described as positions of search agent and $\overrightarrow{M_S}$ can be described as search agent movements.

The characteristics of b is randomized that is responsible for enable balanicng among exploitation and exploration. b is computed as follows,

$$b = 2 \times a^2 \times \mathrm{RD} \tag{4}$$

where RD can be described as random number which presented in the variety of [0, 1].

Near to the best search agent

Finally, search agent is updating location with related to optimal search agent. The best search agent position is presented as follows,

$$\overrightarrow{D_S} = \left|\overrightarrow{c_S} + \overrightarrow{M_S}\right| \tag{5}$$

where, $\overrightarrow{D_S}$ can be described as distance among the best-fit search agent and search agent (optimal seagull whose fitness parameter is less).

Attacking

The exploitation goal is to feat the knowledge in addition past of the search process. Seagulls can variation the angle of attack in addition to speed continuously during migration [24]. It manages their altitude by using weight and wings. During attacking the prey, the movement of spiral characteristics happens in the air. These characteristics of X, Y, Z planes are presented as follows,

$$X^{`} = R \times \mathrm{Cos}(K) \tag{6}$$

$$Y^{`} = R \times \mathrm{Sin}(K) \tag{7}$$

$$Z^{`} = R \times K \tag{8}$$

$$R = U \times E^{\mathrm{KV}} \tag{9}$$

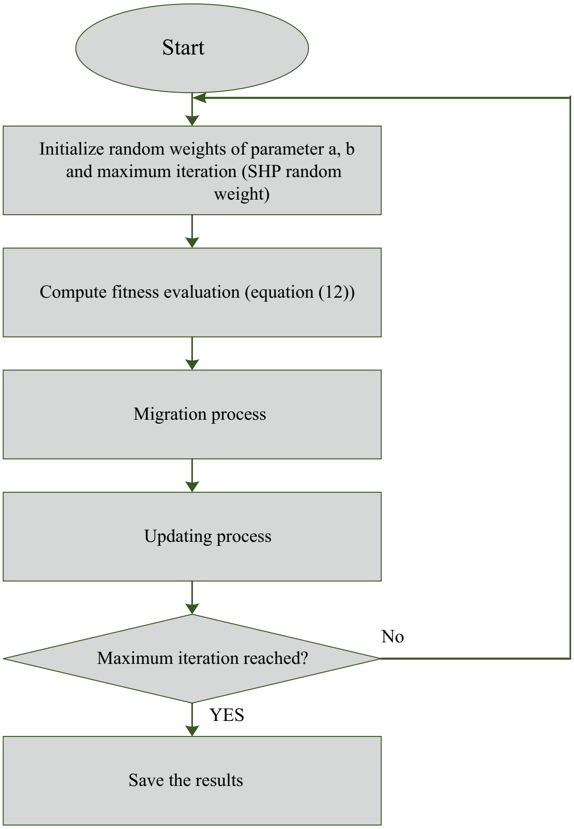

Start
Initialize random weights of parameter a, b and maximum iteration (SHP random weight)
Compute fitness evaluation (equation (12))
Migration process
Updating process
Maximum iteration reached?
No
YES
Save the results

Fig. 2 Flowchart of the proposed methodology

where U and V can be described as constants value that define the spiral shape, K can be described as chance amount in variety $[0 \leq K \leq 2\pi]$, R can be defined as range of every turn of the twisting.

$$\overrightarrow{p_S}(X) = \left(\overrightarrow{D_S} \times X^{`} \times Y^{`} \times Z^{`}\right) + \overrightarrow{p_{\mathrm{bS}}}(X) \tag{10}$$

where, $\overrightarrow{p_S}(X)$ stores the optimal answer, in addition, inform the location of remaining search agents. The SOA algorithm is initiated with a haphazard created population. The search agents have updated their locations with related toward optimal search agent in the repetition procedure. Additionally, a is linearly reduced from F_C to 0. In the flat change among exploitation and examination, parameter b is accountable. So, the SOA is taken as optimization due to its optimal exploitation and examination capability. The flowchart of the projected methodology is demonstrated in Fig. 2.

```
Algorithm 1:
Pseudocode of proposed methodology
Input: Numerous handover process
Output: Optimal handover process
SHP: Ranking network
Formulation of a pair-wise comparison matrix
$S_{\mathrm{HP}} = \begin{bmatrix} A_{11} & \ldots & A_{1N} \\ \ldots & \ldots & \ldots \\ A_{M1} & \ldots & A_{\mathrm{MN}} \end{bmatrix}$
Computation of the weights for matrix consistency by using SOA
SOA: Weight Selection
Initialization of random weights
Compute fitness function by Eq. (12)
Migration process (1–5)
Updating Process (6–10)
Check the termination condition
Save the optimal weights
SHP: process continued
Normalized context matrix
$\mathrm{NCM} = \begin{bmatrix} \widetilde{A_{11}} & \ldots & \widetilde{A_{1N}} \\ \ldots & \ldots & \ldots \\ \widetilde{A_{M1}} & \ldots & \widetilde{A_{\mathrm{MN}}} \end{bmatrix}$
Ranking the network
Select the initial one
Enable optimal handover process
```

Initially, random weights are utilized in SHP model for optimal handover process in heterogeneous network. The optimal weights are selected with the help of SOA algorithm. In this paper, the weights can be achieved by resolving the optimization issue. The maximization function is considered toward fitness function which is summary of the complete parameter of the position variations among networks. The computation of the weights of the parameters are achieved with the help of SOA algorithm. To design a SOA method, the ranking value of network (N), network (I), and assumption of number of networks (M) and attributes (N). The objective function of the proposed methodology is presented as follows,

$$\Delta = \sum_{R=1}^{M} \sum_{S=R+1}^{M} |N_R - N_S| \tag{11}$$

In case of SHP,

$$\Delta_{SHP} = \sum_{R=1}^{M} \sum_{S=R+1}^{M} |W(1)N_R - W(2)N_S| \tag{12}$$

where, $W(1)$ and $W(2)$ can be described as weight parameters which optimally selected by SOA. The optimal selection of weight factors is maximizing the fitness function. By the way, the optimal handover process is enabled in the heterogeneous network which improved the quality of the communication.

4 Result and Discussion

The presentation of the projected technique is evaluated in this section. To authenticate the proposed methodology, the projected method can be simulated in NS2 in addition performances are analyzed. The evaluation of the projected method can be evaluated with the assistance of performance metrics such as network lifetime, energy consumption, delivery ratio, throughput, call drop and delay. The efficiency of the projected methodology can be assessed by comparison analysis. The projected technique can be contrasted with the existing techniques like PSO and GA. Three different modules are considered to evaluate the proposed methodology's performance. The simulation values of the proposed method are presented in Table 2.

- **First module**—Simulation for non-overlapped cell areas
- **Second module**—Simulation for partially overlapped cell areas
- **Third module—**Simulation for fully-overlapped cell areas

The performance metrics are utilized to assess the recital of the projected methodology which is presented in following,

Delay: Delay is defined as the time variation among receiver received packet and sending a packet. When computing delay, complete delay parameters should be considered such as queuing delay, processing delay, transmission delay, and propagation delay.

Table 2 Implementation parameters

S. No	Description	Value
1	Initial idle power	0.035
2	Initial receive power	0.395
3	Initial transmit power	0.660
4	Initial energy	10
5	Simulation time	500 s
6	Dimension of X	1000 m
7	Dimension of Y	1000 m
8	Antenna	Omni antenna
9	MAC Protocol	802_11
10	Propagation	Two-way rounds
11	Channel	Wireless channel

$$\text{Delay} = \sum_{N=1}^{M} (r_N - s_N) \tag{13}$$

Call Drop: The information drops at the end node (receiver) during communication.

Energy consumption: Energy consumption is defined as the total energy consumed by the network to perform the communication in heterogeneous network.

Latency: It can be defined as the amount of time considered for a packet of data to be processed, transmitted, and connected with the mobile nodes.

Network Lifetime: It can be defined as overall lifetime of the networks during transmitter and receiver.

Throughput: It can be defined as efficient delivery of packets or messages to the receiver with the help of communication channel. The throughput is computed based on below equation,

$$\text{Throughput} = \frac{\text{Number of bits reached at the destination}}{1000}$$

The network creation, handover initialization, and handover process are illustrated in Figs. 3, 4 and 5. Initially, the ranking of networks is achieved by SHP. After that, the optimal weight parameters are selected with the help of SOA algorithm. Based on the proposed methodology, the efficient network connection is selected and its connected to the node. The optimal handover process is enabled by proposed SHP and SOA algorithm. The quality of the communication is enabled by proposed methodology. To validate the proposed methodology, three modules are considered which are presented in below section.

4.1 First Module—Simulation for Non-Overlapped Cell Areas

In this module, the network is created with non-overlapped cell areas and analyzed the proposed methodology. During this module, the proposed methodology is enabled the optimal communication. The performance metrics are computed for justify the proposed methodology and it is compared with the existing methods. The delay in the module 1 is illustrated in Fig. 6. From Fig. 6, the proposed methodology has been achieved 5 s and PSO and GA has been achieved 6 s and 8 s, respectively. Thus, the proposed methodology is achieved low delay in the communication system. The call drop of the module 1 is illustrated in Fig. 7. From Fig. 7, the proposed methodology has been achieved at 0.23 s, and PSO and GA has been achieved at 0.3 s and 0.4 s, respectively. Thus, the proposed methodology is achieved low call drop in the communication system (Table 3).

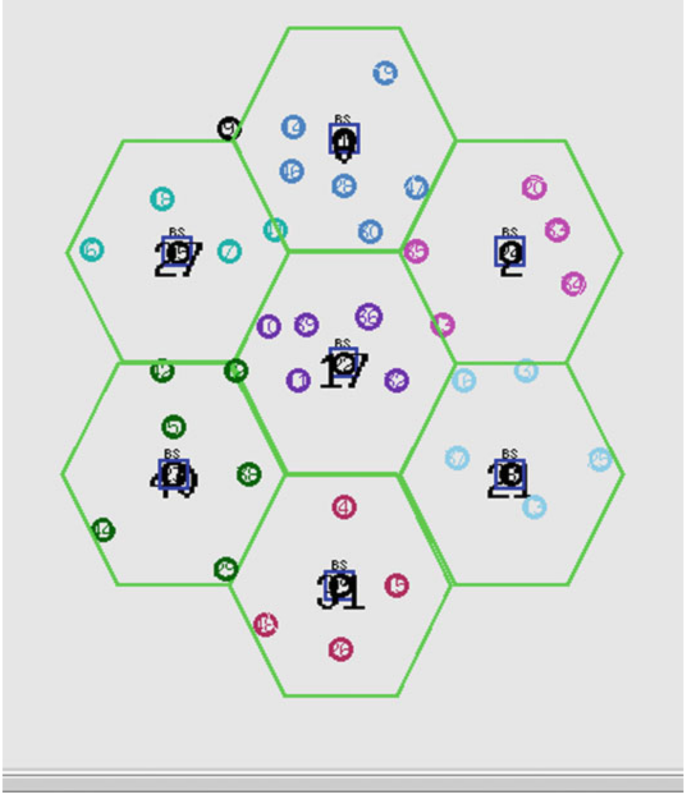


Fig. 3 Network creation

The energy consumption of the module 1 is illustrated in Fig. 8. From Fig. 8, the projected technique is achieved at 0.14 J, and PSO and GA has been achieved 0.2 J and 0.9 J, respectively. Thus, the proposed methodology is achieved low energy consumption in the communication system. The latency of the module 1 is illustrated in Fig. 9. From Fig. 9, the proposed methodology has been achieved at 0.012 ms and PSO and GA has been achieved at 0.02 ms and 1.02 ms, respectively. Thus, the projected technique is achieved low latency in the communication system. The network lifetime of the module 1 is illustrated in Fig. 10. From Fig. 10, the proposed methodology has been achieved 62 s and PSO and GA has been achieved 61 s and 60 s, respectively. Thus, the proposed methodology is achieved high network lifetime in the communication system. The throughput of the module 1 is illustrated in Fig. 11. From Fig. 11, the proposed methodology has been achieved 1452 kbps and PSO and GA has been achieved 1325 kbps and 1225 kbps, respectively. Thus, the proposed methodology is achieved high throughput in the communication system. The delivery ratio of the module 1 is illustrated in Fig. 11. From Fig. 11, the proposed methodology has been achieved 0.95 and PSO and GA has been achieved at 0.89 and

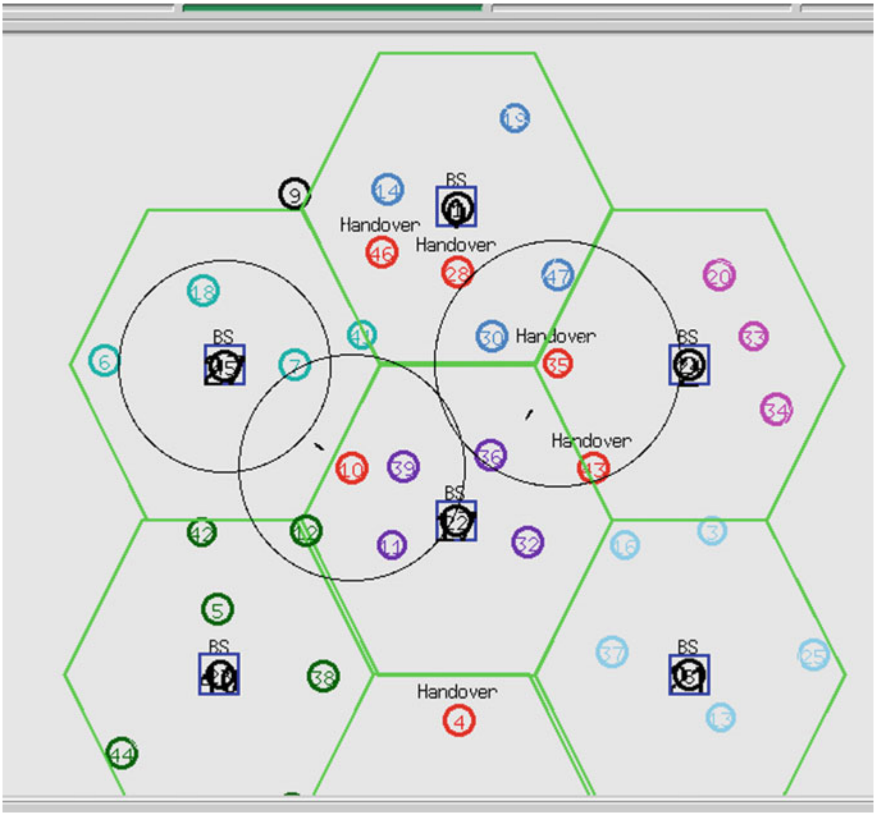


Fig. 4 Initial handover process

0.85, respectively. Thus, the proposed methodology is achieved high delivery ratio in the communication system (Fig. 12).

4.2 *Second Module—Simulation for Partially Overlapped Cell Areas*

In this module, the network is created with partially overlapped cell areas and analyzed the proposed methodology. The hexagon cells are overlapped partially with the users in the heterogeneous network. In this module, the proposed methodology is achieved the optimal handover process which reduces the delay and enhances the efficient communication. The proposed methodology is validated with the help of comparison analysis. The proposed method is compared with the conventional methods such as PSO and GA (Figs. 13, 14, 15, 16, 17, 18 and 19).

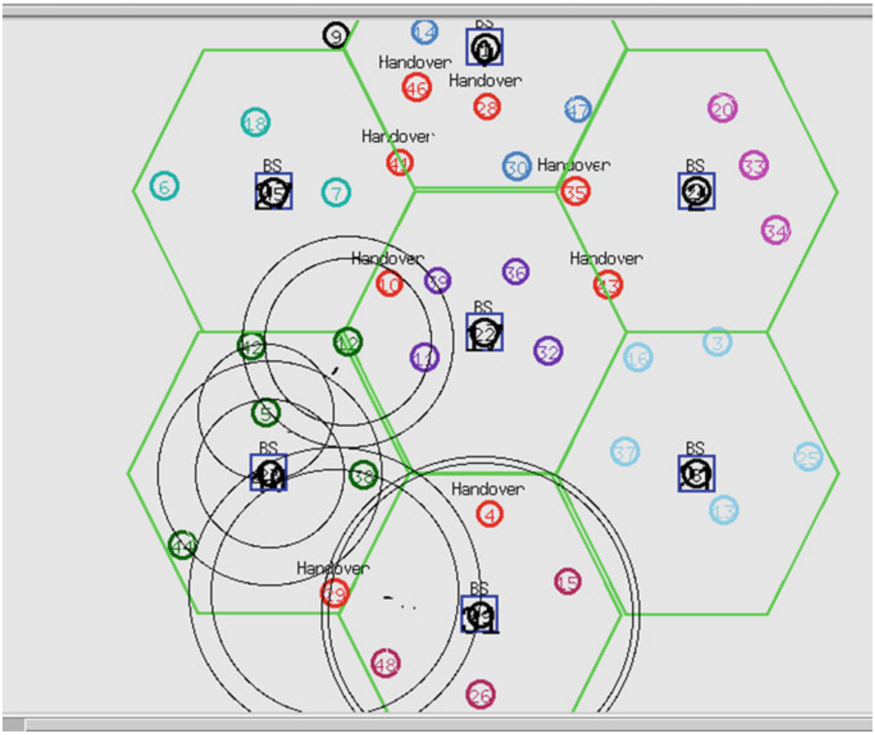


Fig. 5 Proposed handover process

Fig. 6 Delay in non overlapped cells

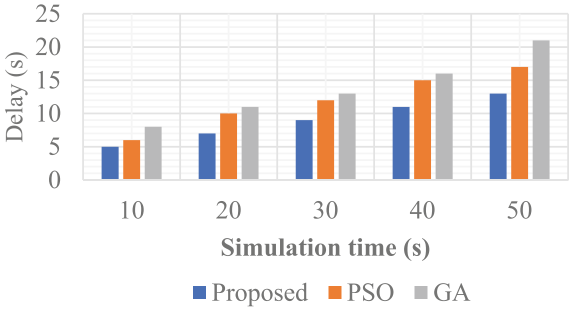


Fig. 7 Call drop in non overlapped cells

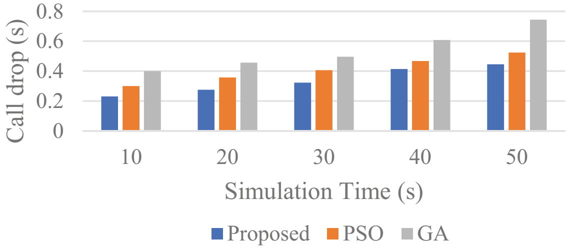


Table 3 Comparison analysis of the proposed method

S. No	Metrics	Proposed	PSO	GA
1	Delay	5 s	6 s	8 s
2	Call drop	0.23 s	0.3 s	0.4 s
3	Energy consumption	0.14 J	0.2 J	0.9 J
4	Latency	0.012 s	0.02 s	1.02 s
5	Network lifetime	62 s	61 s	60 s
6	Throughput	1452 kbps	1325kpbs	1225kpbs
7	Delivery ratio	0.95	0.89	0.85

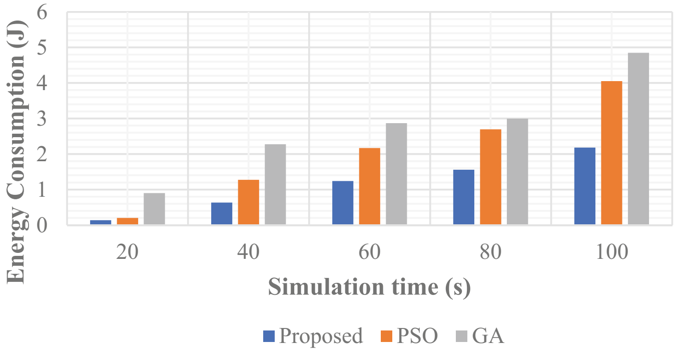


Fig. 8 Energy consumption in non overlapped cells

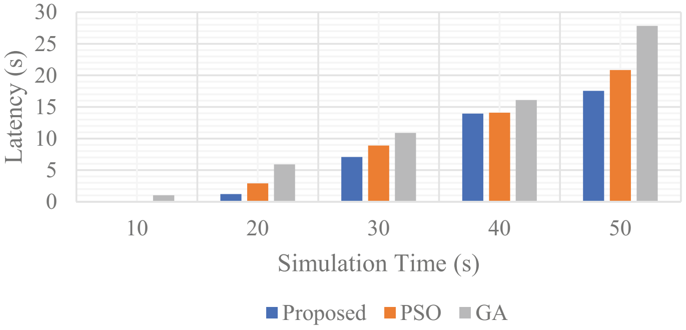


Fig. 9 Latency in non overlapped cells

In this module analysis, the projected methodology has been attained best results of presentation metrics such as delay, delivery ratio, energy consumption, network lifetime, latency, and call drop.

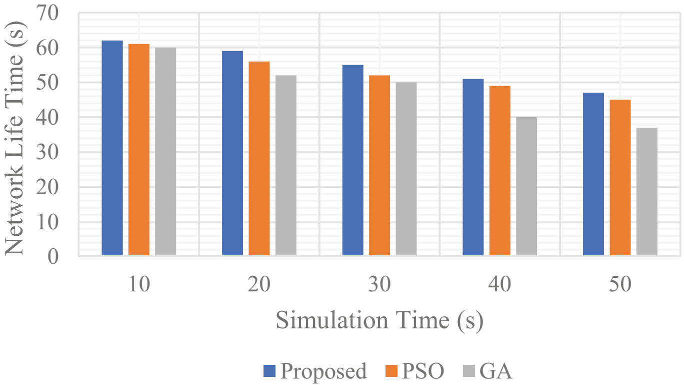


Fig. 10 Network lifetime in non overlapped cells

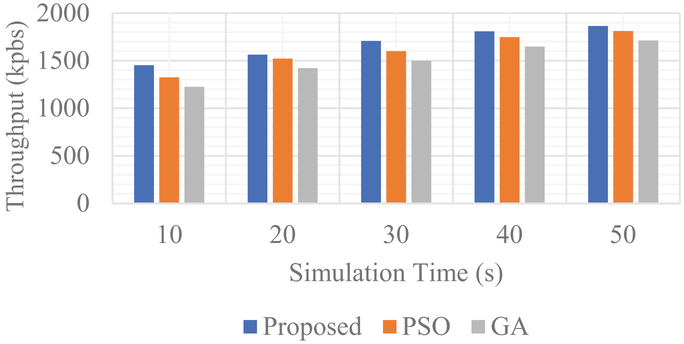


Fig. 11 Throughput in non overlapped cells

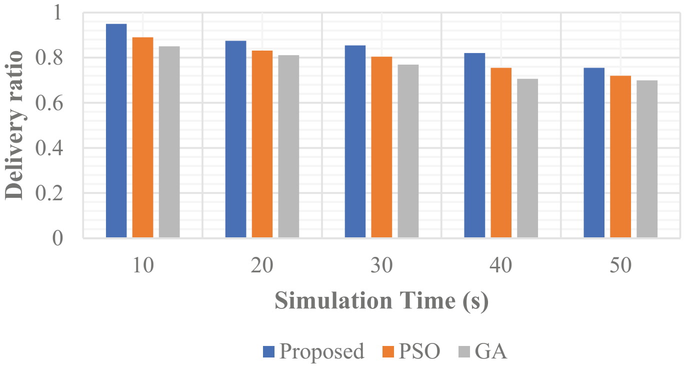


Fig. 12 Delivery ratio in non overlapped cells

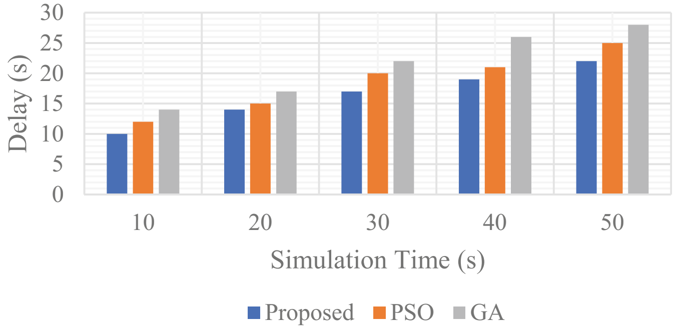


Fig. 13 Delay in partially overlapped cells

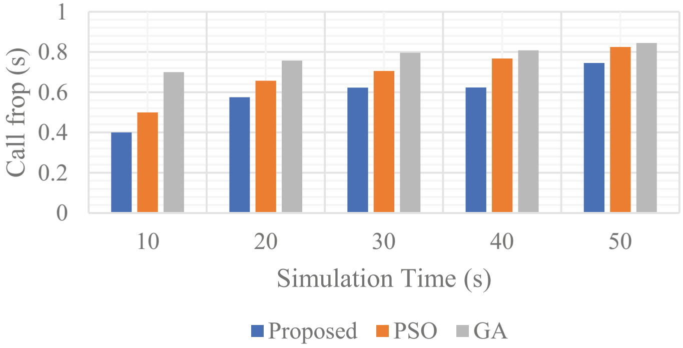


Fig. 14 Call drop in partially overlapped cells

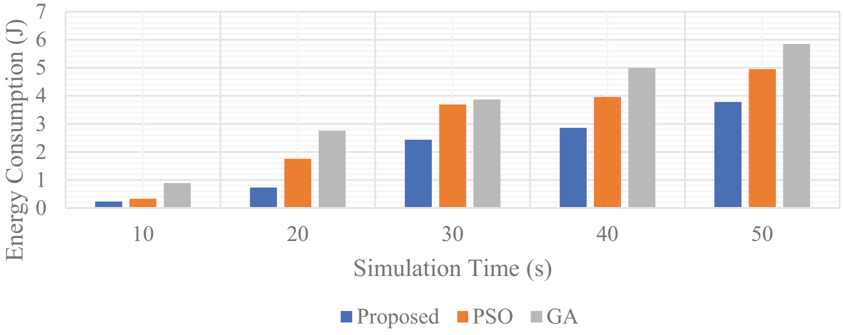


Fig. 15 Energy consumption in partially overlapped cells

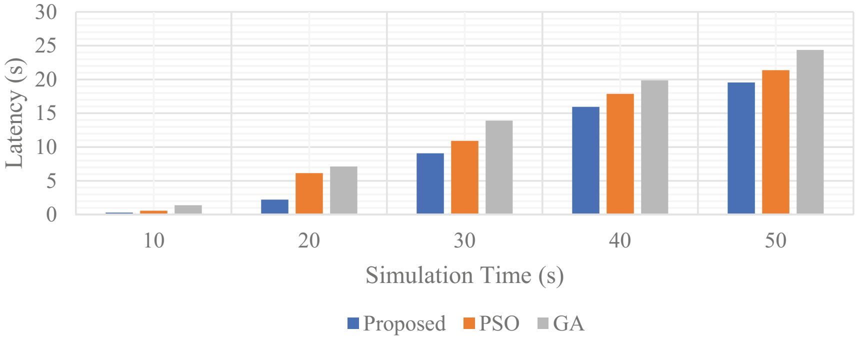


Fig. 16 Latency in partially overlapped cells

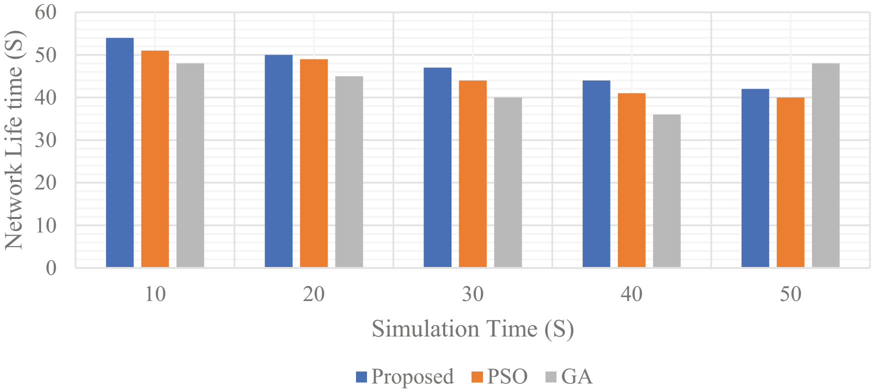


Fig. 17 Network lifetime in partially overlapped cells

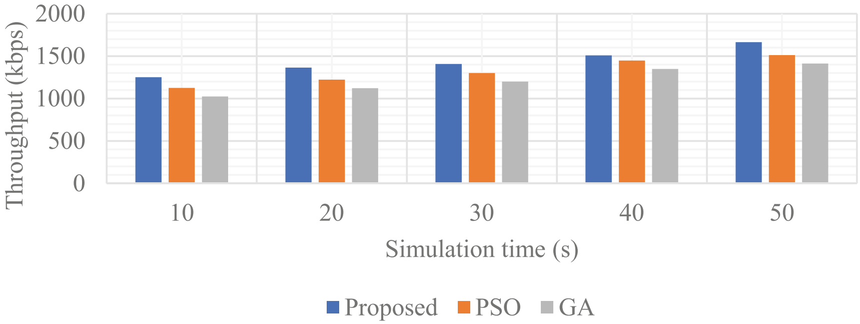


Fig. 18 Throughput in partially overlapped cells

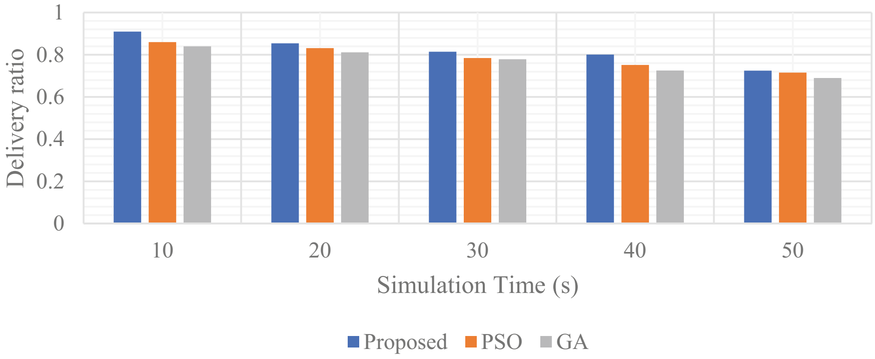


Fig. 19 Delivery ratio in partially overlapped cells

4.3 Third Module—Simulation for Fully-Overlapped Cell Areas

In this module, the network is created with fully-overlapped cell areas and analyzed the proposed methodology. In this condition, the hexagon cells are completely overlapped with the consideration of users in the heterogeneous networks. The proposed methodology is evaluated with the consideration of comparison analysis. The proposed methodology is compared with the conventional methods such as PSO and GA, respectively (Figs. 20, 21, 22, 23, 24, 25 and 26).

In this module analysis, the projected methodology has been achieved best results of presentation metrics such as delay, delivery ratio, energy consumption, network lifetime, latency, and drop.

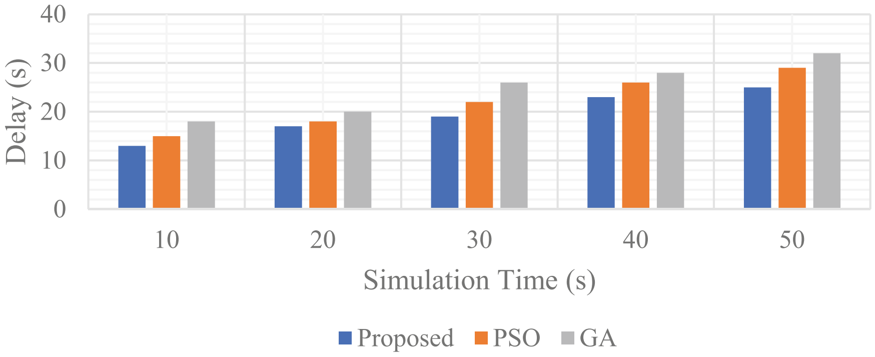


Fig. 20 Delay in fully overlapped cells

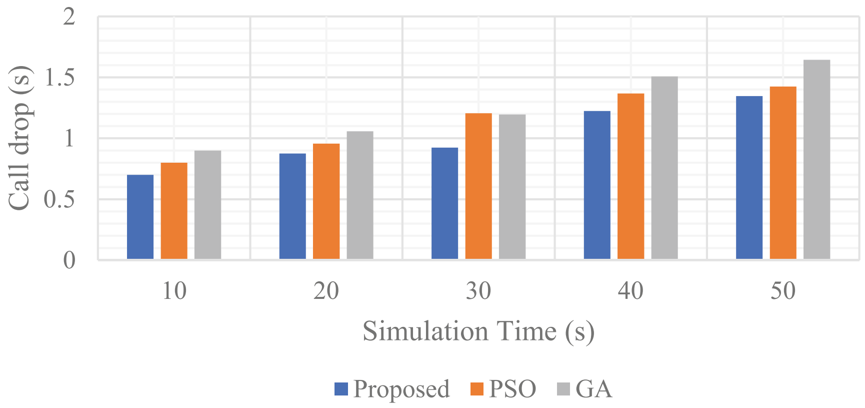


Fig. 21 Call drop in fully overlapped cells

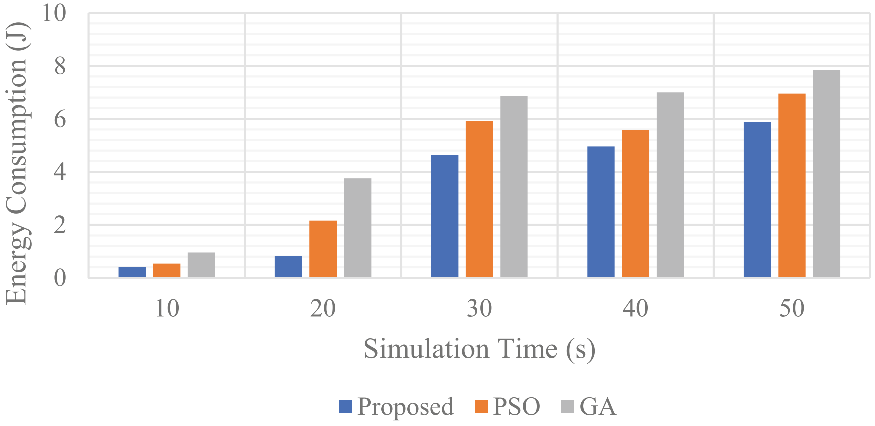


Fig. 22 Energy consumption in fully overlapped cells

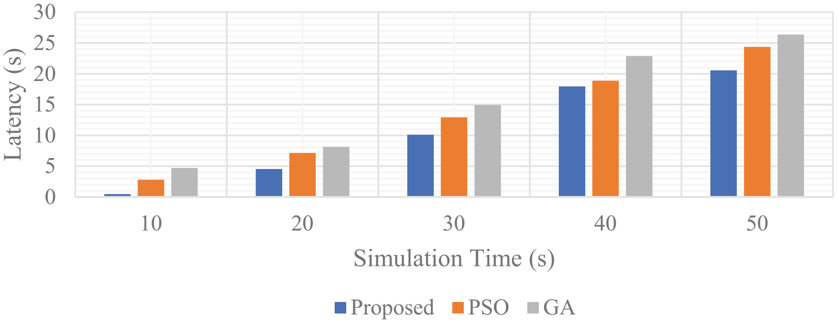


Fig. 23 Latency in fully overlapped cells

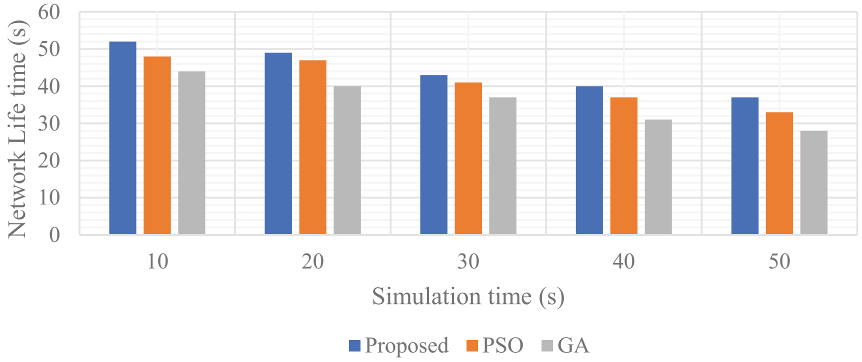


Fig. 24 Network lifetime in fully overlapped cells

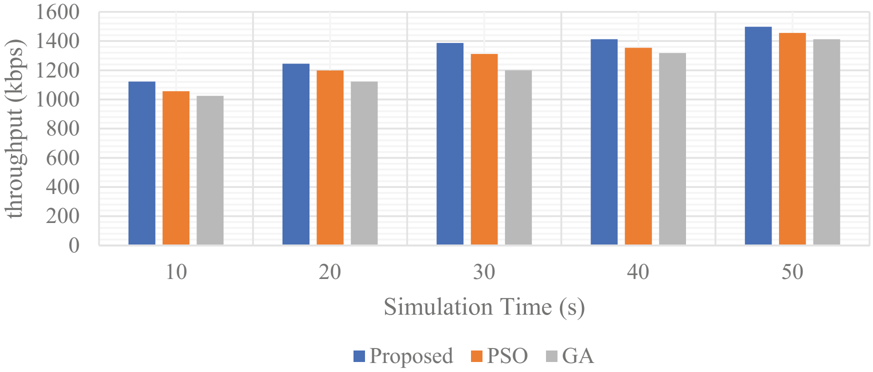


Fig. 25 Throughput in fully overlapped cells

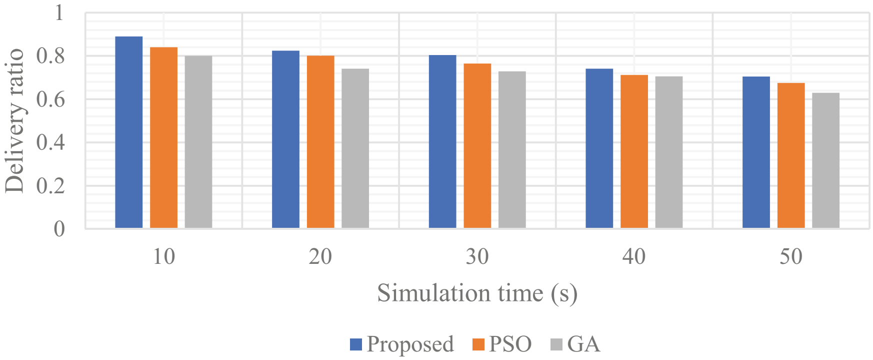


Fig. 26 Delivery ratio in fully overlapped cells

5 Conclusion

In this paper, SHP and SOA has been proposed to enable proper handover in heterogeneous network. The efficient and quality communication is achieved by using proposed SHP and SOA technique. The proposed methodology is utilized to select optimal network for enabling the efficient handover process which enhance the network performance. The proposed methodology is a combination of SHP and SOA. In the SHP, the SOA is utilized to select optimal weighting parameters which enhance the efficiency of the system. Finally, the proposed methodology is achieved efficient handover process which achieved high efficiency. The proposed methodology is implemented in NS2 and performances are evaluated with the help of performance metrics such as latency, network lifetime, energy consumption, throughput, delivery ratio, drop and delay. The proposed methodology is compared with the conventional methods like GA and PSO. The projected methodology has been evaluated with the three modules such as First module—Simulation for non-overlapped cell areas, Second module—Simulation for partially overlapped cell areas, and Third module—Simulation for fully-overlapped cell areas. From the three modules of analysis, the projected methodology has been achieved best results in terms of performance metrics. The proposed method has achieved the performance metrics such as energy consumption: 0.14 J, latency: 0.012 s, network lifetime: 62 s, throughput: 1452 kbps and delivery ratio: 0.95 s In future, the efficient handover technique will be developed for optimal communication with the consideration of Signal to Noise Ratio (SINR).

References

1. Dhand P, Mittal S, Sharma G (2021) An intelligent handoff optimization algorithm for network selection in heterogeneous networks. Int J Inf Technol, pp 1–12
2. Guo D, Tang L, Zhang X, Liang Y-C (2020) Joint optimization of handover control and power allocation based on multi-agent deep reinforcement learning. IEEE Trans Veh Technol 69(11):13124–13138
3. Naresh M, Reddy DV, Reddy KR (2021) Vertical handover in heterogeneous networks using WDWWO algorithm with NN. Int J Electron, pp 1–22
4. Lahby M, Essouiri A, Sekkaki A (2019) A novel modeling approach for vertical handover based on dynamic k-partite graph in heterogeneous networks. Digital Commun Netw 5(4):297–307
5. Alhabo M, Zhang L, Nawaz N, Al-Kashoash H (2019) Game theoretic handover optimisation for dense small cells heterogeneous networks. IET Commun 13(15):2395–2402
6. Zaheeruddin PM (2020) Optimized handoff algorithm for heterogeneous networks. IETE Technical Rev, pp 1–9
7. Haldorai A, Kandaswamy U (2019) Cooperative spectrum handovers in cognitive radio networks. In: Intelligent spectrum handovers in cognitive radio networks, pp 1–18. Springer, Cham
8. Elechi P, Orike S, Akujobi EC (2021) Minimization of handoff failure in a heterogeneous network environment using multi criteria fuzzy system. J Telecommun Electr Comput Eng (JTEC) 13(2):17–22

9. Beshley NK, Yaremko O, Beshley H (2021) A self-optimizing technique based on vertical handover for load balancing in heterogeneous wireless networks using big data analytics. Applied Sci 11(11):4737
10. Kumar S (2021) Efficiency evaluation of handover management techniques in LTE heterogeneous networks. In: International conference on artificial intelligence and sustainable computing, pp 121–134. Springer, Cham
11. Souza DDS, Vieira RF, Seruffo MCDR, Cardoso DL (2019) A novel heuristic for handover priority in mobile heterogeneous networks. IEEE Access 8:4043–4050
12. Patil MB, Patil R (2021) Fuzzy based network controlled vertical handover mechanism for heterogeneous wireless network. Mater Today: Proc
13. He-Wei Yu, and Biao Zhang. A heterogeneous network selection algorithm based on network attribute and user preference. AD hoc Netw 72: 68-80
14. Silva KC, Becvar Z, Cardoso EHS, Francês CRL (2018) Self-tuning handover algorithm based on fuzzy logic in mobile networks with dense small cells. In: 2018 IEEE wireless communications and networking conference (WCNC), pp 1–6. IEEE
15. Khan M, Din S, Gohar M, Ahmad A, Cuomo S, Piccialli F, Jeon G (2017) Enabling multimedia aware vertical handover management in internet of things based heterogeneous wireless networks. Multimedia Tools Appl 76(24):25919–25941
16. Beshley M, Kryvinska N, Yaremko O, Beshley H (2021) A self-optimizing technique based on vertical handover for load balancing in heterogeneous wireless networks using big data analytics. Appl Sci 11(11):4737
17. Naresh M, Reddy DV, Reddy KR (2020) Multi-objective emperor penguin handover optimisation for IEEE 802.21 in heterogeneous networks. IET Commun 14(18):3239–3246
18. Alhammadi A, Roslee M, Alias MY, Shayea I, Alquhali A (2020) Velocity-aware handover self-optimization management for next generation networks. Applied Sci 10(4):1354
19. Al Emam FA, Nasr ME, Kishk SE (2020) Coordinated handover signaling and cross-layer adaptation in heterogeneous wireless networking. Mobile Netw Appl 25(1):285–299
20. Almutairi AF, Hamed M, Landolsi MA, Algharabally M (2018) A genetic algorithm approach for multi-attribute vertical handover decision making in wireless networks. Telecommun Syst 68(2):151–161
21. Almutairi AF, Al-Gharabally M, Salman AA (2021) Particle swarm optimization application for multiple attribute decision making in vertical handover in heterogeneous wireless networks. J Eng Res 9(1)
22. Dhiman G, Singh KK, Soni M, Nagar A, Dehghani M, Slowik A, Kaur A, Sharma A, Houssein EH, Cengiz K (2021) MOSOA: a new multi-objective seagull optimization algorithm. Expert Syst Appl 167:114150
23. Dhiman G, Singh KK, Slowik A, Chang V, Yildiz AR, Kaur A, Garg M (2021) EMoSOA: a new evolutionary multi-objective seagull optimization algorithm for global optimization. Int J Machine Learn Cybern 12(2):571–596
24. Panagant N, Pholdee N, Bureerat S, Kaen K, Yıldız AR, Sait SM (2020) Seagull optimization algorithm for solving real-world design optimization problems. Mater Testing 62(6):640–644
25. Duong TM, Kwon S (2020) Vertical handover analysis for randomly deployed small cells in heterogeneous networks. IEEE Trans Wireless Commun 19(4):2282–2292
26. Zaheeruddin PM (2020) Optimized handoff algorithm for heterogeneous networks. IETE Technical Rev, pp 1–9.

Reconstructing Medical Images Using Generative Adversarial Networks: A Study

Phenilkumar Buch and Amit Thakkar

Abstract Generative adversarial networks (GANs) have been studied and utilized as an alternative to solve medical imaging problems. Major medical imaging applications include image reconstruction, denoising, segmentation, data augmentation, anomaly detection, and synthesis using image-to-image translation (I2I) techniques. There have been many notable improvements for GANs recently, and therefore, a review of notable advances when applying GANs for medical image reconstruction has been conducted for this paper. The aim of this paper is to introduce key I2I ideas and algorithms that work for medical image reconstruction applications. This study presents a review of various GAN architectures and loss functions used for medical image reconstruction that has not been done before.

Keywords Generative adversarial networks · Medical imaging · Image reconstruction

1 Introduction

Deep learning techniques have been increasingly embraced by researchers because of the rise in availability of medical image data. In the area of medical imaging, generative adversarial networks (GANs) have many potential applications, and these networks have revolutionized the entire field.

GANs are a category of deep generative model based on deep learning algorithms introduced in 2014 by Goodfellow et al. [1]. GANs learn the distributions of input

P. Buch (✉)
Computer Engineering Department, Devang Patel Institute of Advance Technology and Research, Charotar University of Science and Technology (CHARUSAT), Changa, Gujarat, India
e-mail: phenil.buch@gmail.com

A. Thakkar
Computer Science and Engineering Department, Chandubhai S Patel Institute of Technology, Charotar University of Science and Technology (CHARUSAT), Changa, Gujarat, India
e-mail: amitthakkar.it@charusat.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_6

data whether it is in the form of text, image, audio, or video. To achieve this GANs employ an adversarial training process.

GANs represent a significant advancement in the area of generative learning, and a GAN typically consists of two separate neural networks. The first is called the discriminator network, and the other is called the generator network. The task of the generator network is to create synthetic data that resembles a data point from the real-data distribution. The generator's input is a random noise vector, and sometimes, an additional condition is also input in case the GAN is being used for conditional generation. The condition may be another vector or an image or a class label. The input of the discriminator is data from the real distribution as well as the synthetic data created by the generator. The output is either 0 or 1 depending on the probability of the input being fake or real. Therefore, the generator aims to make sure that the discriminator outputs a high probability of the discriminator's input being real when the input is actually fake data that is synthetic. The learning process becomes a min–max game inspired by Game Theory. The D loss and G loss are combined to form "adversarial loss" as shown in Eq. (1) in Sect. 2. Figure 1 shows the basic GAN architecture.

There have been many GAN variants developed after the initial variant which is popularly referred to as Vanilla GAN. Some variants modify the objective function of the discriminator while others modify the generator objective. Many variants introduce architectural changes including adding or removing certain loss functions used. There are hundreds of GAN variants, and a comprehensive review of GANs can be found in [2], but this paper focuses only on the GAN variants which have been used for I2I of medical images.

Medical image reconstruction can be interpretated as an I2I task. Medical images obtained in clinics and hospitals are contaminated with noise and artifacts, and the removal of these is the task of medical image reconstruction. This work reviews work that deal with the image modalities of CT scans (computer tomography), MRI

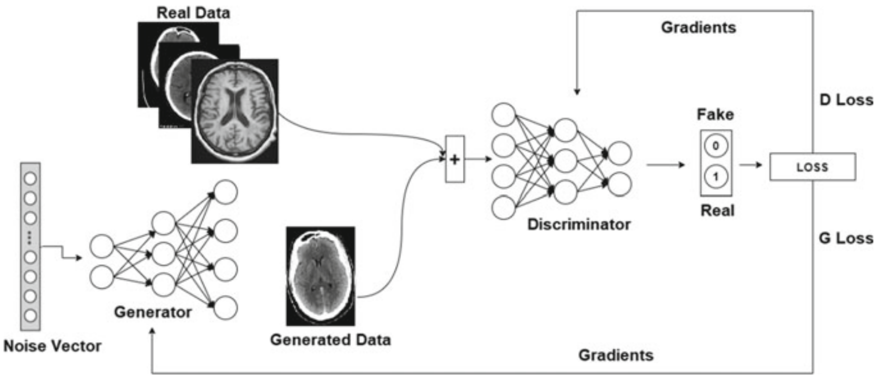


Fig. 1 Basic GAN architecture

scans (magnetic resonance imaging), PET scans (positron emission tomography), ultrasound, X-rays, and more.

The rest of this paper is outlined as follows—Sect. 2 provides a short introduction to I2I GAN variants. Section 3 presents a review of various works related to the medical imaging task of image reconstruction that use GANs. Finally, in Sect. 4, this paper introduces the issues that GANs face associated with medical imaging tasks and some research gaps.

2 Image-To-Image Translation Using GANs

A majority of I2I GAN models, if not all, build and improve upon deep convolutional GAN (DCGAN) [3]. DCGAN is a milestone GAN variant in which convolutional neural networks (CNNs) are used as the constituent networks of the GAN instead of fully-connected multi-layer perceptron neural networks as used in Vanilla GAN. As per observations made during the writing of this paper, DCGANs are considered as the base GAN variant for computer vision applications instead of Vanilla GANs. In DCGANs, there are no pooling layers in the generator and discriminator networks like those found in regular CNNs. The discriminator uses convolution operations on its input, whereas the generator uses de-convolution operations. Each internal layer of the generator and discriminator has batch normalization [4] so that network training is stabler and faster. The Adam optimizer [5] is utilized by DCGANs to minimize/maximize the GAN objective.

I2I tasks are those where a deep learning model learns to translate images belonging to two or more image domains or modalities. An example of such a task would be translating between black-and-white images and colored images. Another example would be converting an image of a cat to an image of a dog that shares the domain-independent features with the input cat image. In this case, one domain would be the set of all cat images, and the other domain would be the set of all dog images. The GAN would essentially be learning how to map one data distribution to another.

There are two scenarios for I2I using deep learning techniques. The first scenario is when there is an availability of a paired image dataset in which there exists a one-to-one mapping of images belonging to two domains of images. In the second scenario, no such mappings exist for the two sets of images within the dataset. Sub-Sects. 2.1 and 2.2 present a very brief overview of I2I for paired and unpaired images. Alotaibi [6] provides an in-depth review of GANs used for I2I.

2.1 Image-to-Image Translation for Paired Images

I2I for paired images is basically supervised I2I since there exists a clear target image for every image belonging to the source image domain. Output generated by the GAN

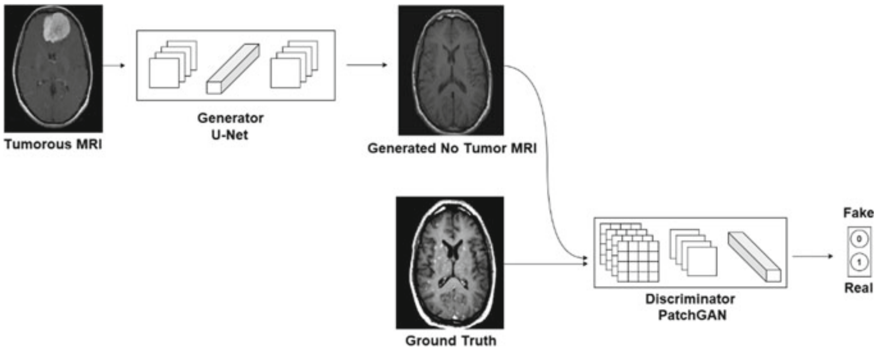


Fig. 2 pix2pix architecture

is supposed to be deterministic in this case. The two most popular GAN variants for paired I2I are pix2pix [7] and StarGAN [8].

Pix2pix is a conditional GAN [9] variant that uses the U-Net [10] architecture for its generator. It also introduces an original PatchGAN architecture that is used for its discriminator network. The U-Net is basically an autoencoder [11] with skip connections. The U-Net generator in pix2pix takes in an image belonging to a source domain, compresses it to form a encoded vector, and then up-samples (or decodes) the encoded vector to generate an image that is supposed to belong to the target domain. The PatchGAN discriminator is implemented as a CNN network that takes a look at patches of the input images, and a feature vector is output that contains the probabilities of each patch being fake or real. Figure 2 shows the architecture of pix2pix.

Pix2pix uses a composite loss function made up of two components, namely adversarial loss (which is used by a majority of GANs) and pixel-distance loss (also called reconstruction loss). The adversarial loss function is based on the binary cross-entropy loss function [12] and is as shown below in Eq. (1). The discriminator maximizes this objective and the generator minimizes it. The pixel-distance loss simply calculates the mean absolute pixel distance of the generated image from the actual target image.

$$\mathcal{L}_{\text{adversarial}}(G) = \mathbb{E}_x\big[\log(D(x))\big] + \mathbb{E}_z[\log(1 - D(G(z))] \tag{1}$$

In Eq. (1), x is real data, z is a random noise vector, $G(z)$ is the generator output, $D(x)$ and $D(G(z))$ represent the discriminator outputs for real and fake inputs, respectively, and $_x$ and $_z$ represent the expectation over all x and z samples, respectively. In case of conditional generation that is used in pix2pix, $D(x)$ becomes $D(x|y)$, and $G(z)$ becomes $G(z|y)$ where y is a condition or a class label depending on what the GAN should generate. [13] mentions that simply using pixel reconstruction loss alone will result in a blurry image because the pixel loss function tends to average multiple possible modalities. Therefore, an adversarial loss term is necessary for higher quality image.

StarGAN is a model that has the capability to perform I2I for multiple domains. Instead of separate networks for every pair of domains, it generalizes learning by providing not just an input image to the generator but also a "mask vector" that contains domain information as a condition. The mask vector is similar to subnet masks in computer networks. In addition to this, StarGAN is also capable of training with multiple datasets and uses a one-hot vector input to represent the dataset that the input image belongs to.

StarGAN not only uses the adversarial loss and reconstruction loss but also uses a domain classification loss term in its loss function. The generator minimizes the domain classification loss to make sure that the fake image it created belongs to the target image domain.

2.2 *Image-To-Image Translation for Unpaired Images*

In reality, very few datasets have source and target image pairs that have a one-to-one mapping defined. Hence, techniques that do not require supervision for I2I have become quite popular. CycleGAN [14] and UNIT [15] are two such models used for unpaired I2I.

CycleGAN uses two GANs working together to achieve I2I without paired image availability. There are two generators denoted in the original paper as G and F. There are two sets of images belonging to domains X and Y and two discriminators denoted as D_x and D_y to represent the domain of images they classify. G translates images from one image domain to another and F does the opposite. CycleGAN defines cycle consistency as the phenomenon where if the output of one generator is input to another generator, then the final output should equal the initial input of the first generator. This can be understood through the below equations.

$$x \rightarrow G(x) \rightarrow F(G(x)) \approx x \tag{2}$$

$$y \rightarrow F(y) \rightarrow G(F(y)) \approx y \tag{3}$$

Besides using two separate adversarial loss terms, one for each generator, CycleGAN uses a cycle-consistency loss term as defined below.

$$\mathcal{L}_{\text{cycle}}(G,\ F) = \mathbb{E}_x[\|F(G(x)) - x\|_1] + \mathbb{E}_y[\|G(F(y)) - y\|_1] \tag{4}$$

Just like pix2pix, CycleGAN also uses PatchGANs for its discriminator networks and an autoencoder variant that has three components, namely the encoder, decoder, and the transformer for its generator. The transformer is made up of residual blocks defined in ResNet [16]. Figure 3 shows the architecture of CycleGAN.

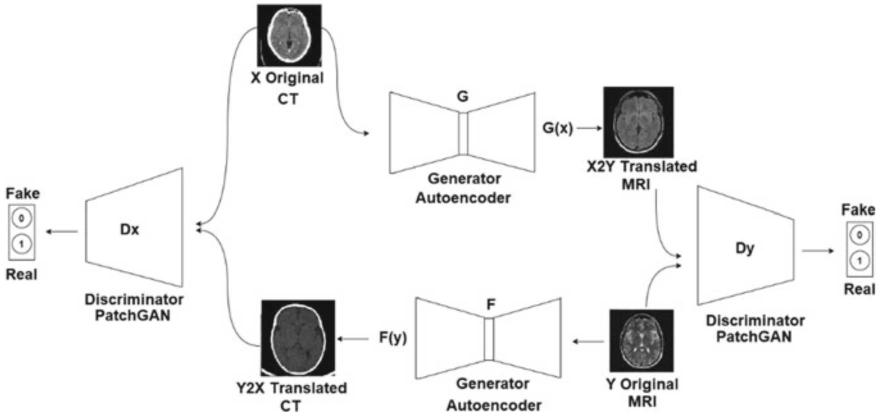


Fig. 3 CycleGAN architecture

3 Applications of GANs in Medical Image Reconstruction

There have been a few papers discussing the usage of deep learning techniques in the analysis of medical images. Shen et al. [17] makes no mention of GANs and [18] makes the claim that at the time of writing in 2017, they found no peer-reviewed papers utilizing GANs for medical image analysis. Singh et al. [19] presents the history and usage of 3D CNNs for 3D medical imaging tasks. Fu et al. [20] reviews the deep learning techniques solely for the application of medical image registration and presents not just GANs but reinforcement learning [21] and other plausible deep learning methods. Wang et al. [22] reviews CNNs in detail and introduces the usage of transfer learning [23] for medical imaging tasks. Sahiner et al. [24] surveys a lot of deep learning techniques not just for medical imaging but also in radiation therapy and it does not focus solely on GANs.

In 2019, an in-depth review on the usage of GANs for medical imaging was presented in [25]. The work focused on giving a general overview of many variants of GANs and reviewed multiple applications in medical imaging including image synthesis, abnormality detection, image registration, image segmentation, image reconstruction, classification, and others. This work presents in depth only those GAN variants which are most commonly used in medical imaging and I2I. This work is specifically a review about image reconstruction using GANs.

Kaji and Kida [26] surveys I2I methods for medical image reconstruction and image synthesis. This work is focused on multiple deep learning techniques besides GANs for image reconstruction and synthesis. Zhao et al. [27] discusses several loss functions for image reconstruction, and restoration and also presents a novel loss function for the same. Table 1 provides a brief summary of important loss functions relevant to GANs in medical I2I applications.

Table 1 Common GAN loss functions

Loss	Summary
Binary cross entropy/log loss	Used in classification machine learning algorithms. If predicted class is closer to actual class, then loss is small in value
Adversarial loss	Main loss that defines a GAN also called min–max loss. Usually based on cross entropy loss but may be based on other losses like least squares loss, hinge loss or Wasserstein loss
Wasserstein loss	Calculated as the minimum distance between two probability distributions (generated and real) based on the earth-mover distance
Least squares loss	Based on L2 loss. Loss is proportional to the distance of a generated sample from real-data distribution
Cycle-consistency loss	When unpaired image data is used for I2I translation, this loss is used to make sure that after transformation by two consecutive generators, the result is similar to original input
Pixel loss	Based on the average per-pixel difference between two images. The per-pixel difference may be squared or absolute
Perceptual loss	Can be pixel loss as above or feature-wise loss. Features of images may be extracted using a separately trained neural network and the difference calculated
Structural similarity loss	A perceptual loss function based on SSIM (structural similarity index) which assumes that pixels in an image are dependent on each other. It focuses on luminance, contrast, and structure

3.1 Denoising, Artifact Removal, and Motion-Correction

Medical image denoising consists of removing noise and artifacts from images acquired from different modalities like CT, MRI, and PET scans. Modalities that use ionizing radiation for image acquisition usually generate noise because of use of low dosage of radiation to ensure patient safety. CT scans sometimes have metal artifacts, and MRI scans have streak artifacts because of small number of radial lines in radial k-space sampling. Radial k-space sampling is a technique used along with MR imaging [28]. As the quantity of radial lines used rises, both, the image acquisition time and image quality increase. Reduction in number of radial lines results in low-quality scan with streaking artifacts. PET imaging requires injection of radiotracers which in high quantity is harmful for patients but in low-quantities results in lower image quality.

When using GAN models like pix2pix for denoising, paired data is often acquired in two ways. One is through adding random noise to high-quality scans which has the drawback of having a different noise distribution than practice. The second way is by performing multiple scans at the same time with different settings like changing the level of radiation dose for CT scans. But image registration, scan alignment issues and the issue of an additional radiation dose to the patient are created when this second method is used. It is a necessity for pix2pix to have completely aligned image pairs in order to perform I2I tasks.

When paired data is obtainable, the work [29] makes use of the pix2pix network to accomplish metal artifact removal from CT ear images. Although the work does not explicitly mention the model's name to be pix2pix. Huang et al. [30] utilizes the pix2pix network with no modifications for removing artifacts from Fourier-domain optical coherence tomography (FD-OCT) images.

The work [31] makes use of CycleGAN and proposes CycleWGAN for PET image denoising even when they have paired data for low-dose PET (LDPET) and full dose PET (FDPET) images. Since they have paired data, they add an extra supervised loss term to the three losses that CycleGAN uses by default. The default losses being adversarial loss, cycle-consistency loss, and identity loss. Using Wasserstein distance in place of cross-entropy as the adversarial loss term makes CycleGAN a CycleWGAN based on Wasserstein GAN (WGAN) [32]. Zhao et al. [33] does something very similar and calls their model S-CycleGAN.

Gu et al. [34] present their work on denoising very low-dose CT images of the heart making use of a model called noise-disentangled CycleGAN in which they propose approximately disentangling noise from CT images using another method before feeding the images to a modified CycleGAN. Ma et al. [35] proposes a cycle structure and illumination constrained GAN (CSI-GAN) based on CycleGAN for enhancement of medical images. The CSI-GAN has two extra loss terms for supervision compared to the regular CycleGAN. The authors introduce illumination loss which aims at improving the overall illumination of the translated output and structure loss which is based on the structural similarity loss (SSIM).

A notable work [36] makes use of WGAN along with a perceptual loss function to achieve low-dose CT denoising. Another notable work that does the same and makes use of paired images is [37] which proposes a high-frequency sensitive GAN (HFSGAN) which uses an inception [38, 39] network as the discriminator and two U-Nets combined as the generator. The work makes use of least squares loss as the adversarial loss instead of cross-entropy loss. Park et al. [40] uses a fidelity-embedded GAN (f-GAN) for CT denoising without paired images. Ma et al. [41] makes use of least squares GAN (LSGAN) [42] and a hybrid loss function that has structural similarity loss as a component to perform CT denoising.

A notable work that performs MRI denoising without using pix2pix or CycleGAN is [43]. It proposes a residual encoder-decoder WGAN (RED-WGAN) along with perceptual similarity loss. Armanious et al. [44] proposes MedGAN with a novel generator structure called CasNet and loss functions inspired from StyleGAN [45], namely style loss and content loss. This model is then applied for various medical imaging tasks like translation from PET image scans to CT images which is also referred to as cross-modality synthesis. Other tasks include correcting MR motion artifacts and denoising PET images. Building upon MedGAN, [46] proposes a novel model for MR motion correction called Cycle-MedGAN and two loss functions called cycle-perceptual loss and cycle-style loss which are feature-based perceptual loss functions.

4 Discussion and Conclusion

It was found that a lot more CycleGAN-based works existed compared to pix2pix-based works. This speaks volumes about the lack of availability of paired data for medical imaging tasks. Most of the times if a paired image model is to be used for tasks such as image reconstruction, then either image synthesis or image registration is done with the assistance of a deep learning technique. Only then would the resultant data be useful.

Besides pix2pix and CycleGAN, not many other GAN variants out of hundreds that exist have been tried out for medical image reconstruction. Models including StarGAN and UNIT mentioned in Sect. 2 have rarely been tested by researchers for medical imaging tasks, and this provides opportunity for more work to be done in this area.

Quantitative measures that quantify the performance of GANs for many tasks including image reconstruction tasks are Fréchet inception distance (FID) [47], inception score [48], and more [49]. In such a scenario, GANs used in medical imaging tasks need to be scrutinized microscopically using multiple measures before they can be implemented for real-world hospitals and clinics.

In case pix2pix or CycleGAN or any other GAN fails to remove artifacts from medical scans during the image reconstruction task, the artifacts may be misdiagnosed as calcifications or lesions. Xu et al. [50] discusses how loss functions like the one used in CycleGAN can generate imaginative features that do not actually exist in the original image after I2I. Therefore, care should be taken that unless the performance of the GAN used is evaluated to be critically credible, GAN generated images be used in conjunction with the original medical images by domain experts when performing patient diagnosis.

For this study, over 30 papers were explored to find out the challenges related to medical image reconstruction using GANs and solutions proposed. GANs require multiple loss functions as constraints for the reconstructed image to stay close to the ground truth. They require a lot a fine tuning and a lot of time to train because of training stability issues. Despite this, GANs still remain the best available method for medical image reconstruction.

In conclusion, this study finds that the research direction taken by many studies for medical image reconstruction using GANs tends to be either minor architectural changes or introduction of novel loss functions.

References

1. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. Adv Neural Inf Process Syst 27
2. Gui J, Sun Z, Wen Y, Tao D, Ye J (2020) A review on generative adversarial networks: algorithms, theory, and applications. arXiv preprint arXiv:2001.06937

3. Radford A, Metz L, Chintala S (2015) Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434
4. Bjorck J, Gomes C, Selman B, Weinberger KQ (2018) Understanding batch normalization. arXiv preprint arXiv:1806.02375
5. Zhang Z (2018) Improved adam optimizer for deep neural networks. In: 2018 IEEE/ACM 26th international symposium on quality of service (IWQoS). pp 1–2
6. Alotaibi A (2020) Deep generative adversarial networks for image-to-image translation: a review. Symmetry 12:1705
7. Isola P, Zhu J-Y, Zhou T, Efros AA (2017) Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1125–1134
8. Choi Y, Choi M, Kim M, Ha J-W, Kim S, Choo J (2018) Stargan: unified generative adversarial networks for multi-domain image-to-image translation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 8789–8797
9. Mirza M, Osindero S (2014) Conditional generative adversarial nets. arXiv pre-print arXiv: 1411.1784
10. Ronneberger O, Fischer P, Brox T (2015) U-net: convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention, pp 234–241
11. Bank D, Koenigstein N, Giryes R (2020) Autoencoders. arXiv preprint arXiv:2003.05991
12. Wang Q, Ma Y, Zhao K, Tian Y (2020) A comprehensive survey of loss functions in machine learning. Annals Data Sci, pp 1–26
13. Pathak D, Krahenbuhl P, Donahue J, Darrell T, Efros AA (2016) Context encoders: Feature learning by inpainting. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2536–2544
14. Zhu J-Y, Park T, Isola P, Efros AA (2017) Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE international conference on computer vision, pp 2223–2232
15. Liu M-Y, Breuel T, Kautz J (2017) Unsupervised image-to-image translation networks. In: Advances in neural information processing systems, pp 700–708
16. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
17. Shen D, Wu G, Suk H-I (2017) Deep learning in medical image analysis. Annu Rev Biomed Eng 19:221–248
18. Litjens G, Kooi T, Ehteshami BB, Setio AAA, Ciompi F, Ghafoorian M, van der Laak JA, van Ginneken B, Sánchez CI (2017) A survey on deep learning in medical image analysis. Med Image Anal 42:60–88
19. Singh SP, Wang L, Gupta S, Goli H, Padmanabhan P, Gulyás B (2020) 3D deep learning on medical images: a review. Sensors 20:5097
20. Fu Y, Lei Y, Wang T, Curran Walter J, Liu T, Yang X (2020) Deep learning in medical image registration: a review. Phys Med Biol 65:20TR01
21. Henderson P, Islam R, Bachman P, Pineau J, Precup D, Meger D (2018) Deep reinforcement learning that matters. In: Proceedings of the AAAI conference on artificial intelligence
22. Wang J, Zhu H, Wang S-H, Zhang Y-D (2021) A review of deep learning on medical image analysis. Mob Networks Appl 26:351–380
23. Tan C, Sun F, Kong T, Zhang W, Yang C, Liu C (2018) A survey on deep transfer learning. In: International conference on artificial neural networks, pp 270–279
24. Sahiner B, Pezeshk A, Hadjiiski LM, Wang X, Drukker K, Cha KH, Summers RM, Giger ML (2019) Deep learning in medical imaging and radiation therapy. Med Phys 46:e1–e36
25. Yi X, Walia E, Babyn P (2019) Generative adversarial network in medical imaging: a review. Med Image Anal 58:101552
26. Kaji S, Kida S (2019) Overview of image-to-image translation by use of deep neural networks: denoising, super-resolution, modality conversion, and reconstruction in medical imaging. Radiol Phys Technol 12:235–248

27. Zhao H, Gallo O, Frosio I, Kautz J (2016) Loss functions for image restoration with neural networks. IEEE Trans Comput Imaging 3:47–57
28. Han Y, Yoo J, Kim HH, Shin HJ, Sung K, Ye JC (2018) Deep learning with domain adaptation for accelerated projection-reconstruction MR. Magn Reson Med 80:1189–1205
29. Wang J, Zhao Y, Noble JH, Dawant BM (2018) Conditional generative adversarial networks for metal artifact reduction in CT images of the ear. In: International conference on medical image computing and computer-assisted intervention, pp 3–11
30. Huang C-M, Wijanto E, Cheng H-C (2021) Applying a Pix2Pix generative adversarial network to a fourier-domain optical coherence tomography system for artifact elimination. IEEE Access 9:103311–103324
31. Zhou L, Schaefferkoetter JD, Tham IWK, Huang G, Yan J (2020) Supervised learning with cyclegan for low-dose FDG PET image denoising. Med Image Anal 65:101770
32. Arjovsky M, Chintala S, Bottou L (2017) Wasserstein generative adversarial networks. In: International conference on machine learning, pp 214–223
33. Zhao K, Zhou L, Gao S, Wang X, Wang Y, Zhao X, Wang H, Liu K, Zhu Y, Ye H (2020) Study of low-dose PET image recovery using supervised learning with CycleGAN. PLoS ONE 15:e0238455
34. Gu J, Yang TS, Ye JC, Yang DH (2021) CycleGAN denoising of extreme low-dose cardiac CT using wavelet-assisted noise disentanglement. Med Image Anal 74:102209
35. Ma Y, Liu Y, Cheng J, Zheng Y, Ghahremani M, Chen H, Liu J, Zhao Y (2020) Cycle structure and illumination constrained GAN for medical image enhancement. In: International conference on medical image computing and computer-assisted intervention, pp 667–677
36. Yang Q, Yan P, Zhang Y, Yu H, Shi Y, Mou X, Kalra MK, Zhang Y, Sun L, Wang G (2018) Low-dose CT image denoising using a generative adversarial network with Wasserstein distance and perceptual loss. IEEE Trans Med Imaging 37:1348–1357
37. Yang L, Shangguan H, Zhang X, Wang A, Han Z (2019) High-frequency sensitive generative adversarial network for low-dose CT image denoising. IEEE access. 8:930–943
38. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1–9
39. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2016) Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2818–2826
40. Park HS, Baek J, You SK, Choi JK, Seo JK (2019) Unpaired image denoising using a generative adversarial network in X-ray CT. IEEE Access 7:110414–110425
41. Ma Y, Wei B, Feng P, He P, Guo X, Wang G (2020) Low-dose CT image denoising using a generative adversarial network with a hybrid loss function for noise learning. IEEE Access 8:67519–67529
42. Mao X, Li Q, Xie H, Lau RYK, Wang Z, Paul Smolley S (2017) Least squares generative adversarial networks. In: Proceedings of the IEEE international conference on computer vision, pp 2794–2802
43. Ran M, Hu J, Chen Y, Chen H, Sun H, Zhou J, Zhang Y (2019) Denoising of 3D magnetic resonance images using a residual encoder–decoder Wasserstein generative adversarial network. Med Image Anal 55:165–180
44. Armanious K, Jiang C, Fischer M, Küstner T, Hepp T, Nikolaou K, Gatidis S, Yang B (2020) MedGAN: medical image translation using GANs. Comput Med Imaging Graph 79:101684
45. Karras T, Laine S, Aila T (2019) A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 4401–4410
46. Armanious K, Jiang C, Abdulatif S, Küstner T, Gatidis S, Yang B (2019) Unsupervised medical image translation using cycle-MedGAN. In: 2019 27th European signal processing conference (EUSIPCO), pp 1–5
47. Cohen JP, Luck M, Honari S (2018) Distribution matching losses can hallucinate features in medical image translation. In: International conference on medical image computing and computer-assisted intervention, pp 529–536

48. Heusel M, Ramsauer H, Unterthiner T, Nessler B, Hochreiter S (2017) Gans trained by a two time-scale update rule converge to a local nash equilibrium. Adv Neural Inf Process Syst 30
49. Salimans T, Goodfellow I, Zaremba W, Cheung V, Radford A, Chen X (2016) Improved techniques for training gans. Adv Neural Inf Process Syst 29:2234–2242
50. Xu Q, Huang G, Yuan Y, Guo C, Sun Y, Wu F, Weinberger K (2018) An empirical study on evaluation metrics of generative adversarial networks. arXiv preprint arXiv:1806.07755

Autonomous Mobile Robot for Inventory Management in Retail Industry

Harsh Parikh, Ishika Saijwal, Nisarg Panchal, and Ankit Sharma

Abstract In recent years, the idea of autonomous vehicles is on pace as some automobile companies have decided to develop their autonomous cars. Autonomous mobile robots (AMR) are currently being used in a variety of intra-logistics operations, such as warehousing, terminals, manufacturing, hospitals, and cross-docks. Their advanced control software and hardware allow autonomous operations in dynamic environments. In this paper, we have implemented a differential drive robot equipped with a depth camera and an RP LIDAR. This robot is capable of autonomous navigation through the warehouse environment by processing the data obtained through the sensors in real time. It can navigate to a particular shelf and then count the no. of cartons present on the shelf and compare it with previous data to give us an idea about the present inventory. It uses deep learning-based object detection models for the detection of cardboard boxes on a shelf.

Keywords Autonomous navigation · SLAM · Inventory · Planning and localization · AMR · Depth camera · Deep learning · Machine learning · Image processing

H. Parikh (✉) · I. Saijwal · N. Panchal · A. Sharma
Institute of Technology, Nirma University, Ahmedabad, India
e-mail: 19bic036@nirmauni.ac.in

I. Saijwal
e-mail: 19bec042@nirmauni.ac.in

N. Panchal
e-mail: 19bee070@nirmauni.ac.in

A. Sharma
e-mail: ankit.sharma@nirmauni.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_7

1 Introduction

AMR is a robot that can not only move through but also understand the environment without being in a fixed predetermined path or under any inspection by a manager. They have a combination of sensors that help them interpret and find out their environment, which facilitates them to find the most efficient path possible for the task. They can navigate by avoiding fixed obstructions (buildings, work stations, racks, etc.) and movable obstructions (people, lift trucks, and debris). Though they are like automated guided vehicles (AGVs) but differ in several important ways. One of the differences is flexibility as AGVs must follow much more determined routes than AMRs. These work collaboratively with operators like sorting and picking operations along with finding the most efficient route to complete each task, whereas AGVs typically don't follow the same. In a distribution center environment like a warehouse or an inventory, these complex technologies are integrated with the control systems of the warehouse, which allow AMR improved flexibility to plan their paths between locations within a facility or warehouse. This results in the robot being able to work within the dynamic environment which is given by most order fulfillment operations.

2 Literature Overview

In [1], a compact 3D camera for mapping, ranging, and guiding vehicles is developed using a dual-aperture mask in a camera lens; it uses a projected laser beam for creating a 360-degree range map that can be used to plan trajectories. Nickerson et al. [2] focus on an autonomous robot for a known environment (ARK) that is capable of carrying out certain tasks in a known industrial environment. Fusion of different frames of sensor data for consistent mapping of the environment is shown in [3]. In this work [4], a new vector polar histogram method is formulated that can detect obstacles through a laser radar and assist with path planning. Bakambu [5] proposes a novel odometry technique using a heading sensor that fuses data from a fiber optic gyro and a simple inclinometer. Pfister et al. [6] show the calculation of displacement by comparing range scans at two different positions in the same environment. In [7], certain parameters like operation time, data transfer rate, target surface properties, and the incidence angle of Sick LMS 200 laser scanner are studied as they affect sensing performance. This paper [8] presents three methods for intelligent fusion of data captured from different sensors. Alberto et al. [9] discuss different types of features and criteria for the selection of a feature for building a topological map of the environment. In [10], work is done in navigating a robot through a gate by detecting it with a laser measuring system, and navigation is done using Global Positioning System (GPS) and inertial navigation system (INS). This paper [11] compares the performance of extended Kalman filter-(EKF)-based simultaneous localization and mapping (SLAM) and the compressed extended Kalman filter-(CEKF)-based SLAM. This paper [12] provides insights and information on recent developments and implementation of SLAM. In [13], faster R-CNN was proposed for the first time for real-time object detection as an improved version of fast R-CNN. YOLO V4 is introduced in [14] as development over YOLO V3. In [15], object detection is performed using a modified YOLO V1 neural network that is faster than YOLO V1. Pal et al. [16] are a study of some major object detection models and architectures that are developed till now.

3 Problem Definition

The robot was to be developed so that it can navigate through the environment autonomously to the point specified by the operator while avoiding obstacles in its way, and it should also be capable to detect and count items on each rack and then calculating how many items were missing by comparing it with previous data.

4 Proposed Method

A 2D map was made using hector SLAM, and the camera output was given at the subscriber end. This enables the user to access real-time data in form of a video. Robot operating system visualization (RViz) gave real-time images from the Intel Realsense depth camera mounted on top of the bot. RViz is a 3D visualization tool from robot operating system (ROS) that allows simulation of a robot using and also provides functions like getting sensor data from the simulation environment. For the detection of boxes, various deep learning-based object detecting models were employed that are described in Sect. 6.

5 Implementation

This robot (Fig. 1) was developed using ROS as it enables data transfer between various sensors, actuators, and the computer while also allowing visualization and simulation. SLAM navigation was used to navigate around the environment while avoiding obstacles in its way.

5.1 SLAM (Simultaneous Localization and Mapping)

Autonomous robots can navigate indoors and outdoors while avoiding obstacles in their environment. SLAM is a technique that makes an autonomous robot navigate outdoors and indoors by avoiding obstacles in its environment and making a map,

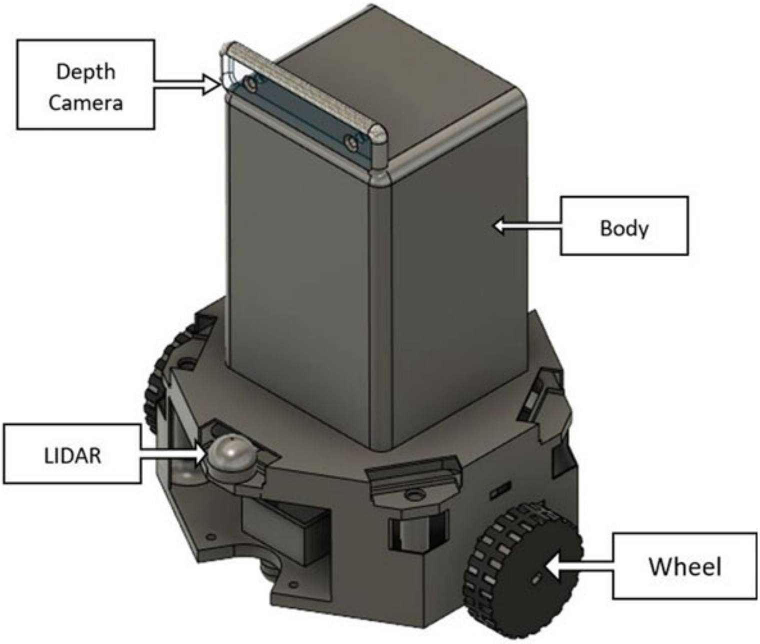


Fig. 1 CAD model of the bot

updating it and also estimating the position of the robot. This algorithm is a blend of mapping, multiple objects, complexity, kinematic modeling, moving objects, sensing, multiple cameras, exploration, loop closure. Range measurement device forms a crucial part that is used for observing the environment by taking different variables into account. For mapping of the environment, we have used GMapping that stands for grid mapping. It uses a particle filter to create grid maps of the environment from the LIDAR data. If there are some noisy and incomplete observations, the particle filter employs approximation updates and prediction to determine the posterior distributions of the states of a Markov process. GMapping requires odometry data (wheel encoder data) for the pose estimation of our robot and laser data (Kinect data) for creating a 2D occupancy grid map. Figure 2 shows the map of its environment created by our bot.

5.2 AMCL (Adaptive Monte-Carlo Localization)

For localization of our robot in the map, we have used AMCL, which is a probabilistic localization system, which is used in robots for 2D movements. In the AMCL approach, a particle filter is utilized to follow the robot's position against a pre-known map. The first step is to generate the environment's map. The robot could begin from

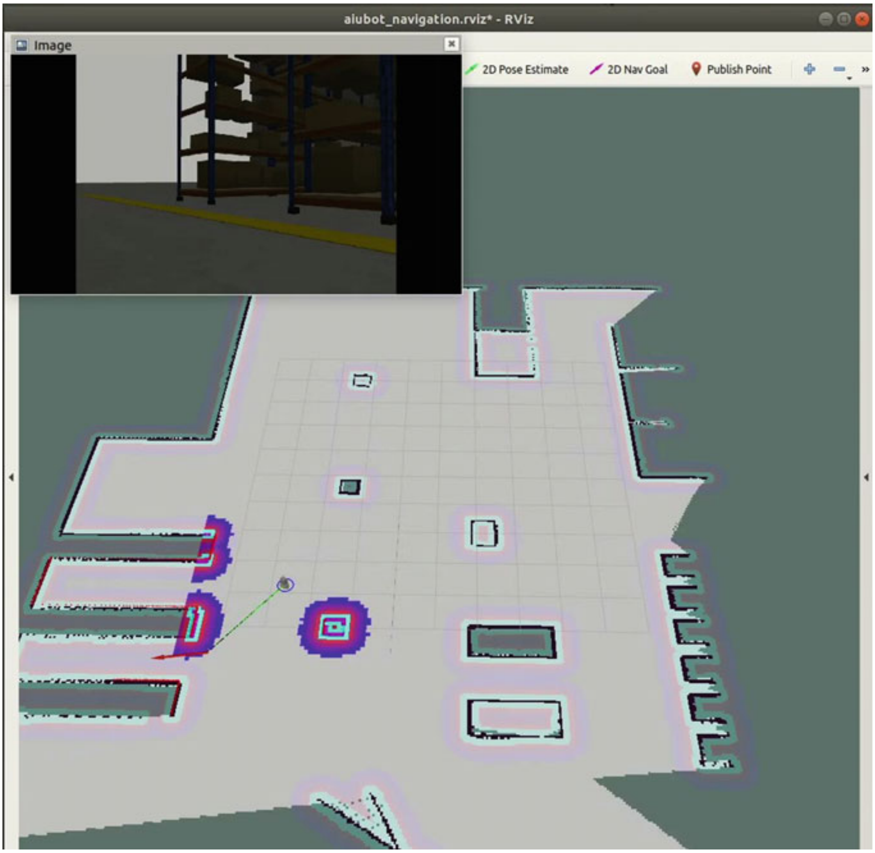


Fig. 2 2D map created by our robot

an initial estimate of its position, or it could be set to a random location. As the robot moves forward, new sample generation starts predicting the robot's position after each command. If the robot losses track of its position, a set of random uniformly distributed samples can be added to recover the robot.

5.3 Localization

Localization estimates the robot's pose and position in accordance with the environment. To localize the map of the environment, the robot laser data and odometry data are necessary. The bot uses adaptive Monte-Carlo localization (AMCL) for localization. For the bot in 2D, it follows a probabilistic localization approach.

5.4 Autonomous Navigation

Once the localization and mapping are completed, autonomous navigation can be performed. ROS packages of navigation stack are used to implement AMCL. The ROS AMCL package provides a node for performing localization on a static map. It subscribes to the previously generated static map, the 2D laser scan data and the TF (Transform) data provided by the robot. TF is a package that allows the user to keep an eye on multiple coordinate frames over-time, it keeps the relationship between coordinate frames in a tree structure buffered in time. The AMCL node publishes the robot's pose and estimated position with respect to the map as generated by TF. This information regarding the obstacles is kept in two costmaps. Global costmaps plan long-term path over the map and is used for global path planning. The local costmaps are employed for local path planning and obstacle avoidance. For visualizing these global and local costmaps, RViz is used. Now, the robot can move autonomously. Here, for assigning a destination goal to the robot in the map, a 2D navigation goal in RViz is given. Now, robot plans the path, and velocity commands are given to the robot controller. Figure 3 shows the path followed by the robot to navigate from its initial position to the desired destination goal assigned.

6 Detection of Boxes

For detection of boxes (only one class) from a shelf, deep learning models like faster R-CNN, Yolo V4-Tiny, Yolo V4 were trained and then tested to choose which perform best for our application. After testing against various real life, Yolo V4 was chosen as it was more robust for detection from unknown data as well as it performed well on the test dataset. The labeled dataset was downloaded from Kaggle https://www.kaggle.com/sakshi12345/cardboard-box-images-with-annotations-for-yolov5. The dataset was split as 70% for training, 20% for testing, and 10% for validation. The dataset had a total of 522 images of resolution 640 × 480. For training of Yolo V4 and Yolo V4-Tiny, the dataset was resized to 416 × 416 before training. After splitting the dataset, there were 366 images in the training directory, 104 images in the test directory, and the remaining 52 images were there in the validation dataset. Before training, various data augmentation techniques were applied like salt and pepper noise, rotation, horizontal, and vertical flip to improve variation in the dataset. The bot was given coordinates of a position such that from there it can see only one shelf at a time, and then, boxes were detected and counted from the image. So, it only had to run inference from a single image, and thus, high frame rates were not required. All the models that were employed are described in brief below.

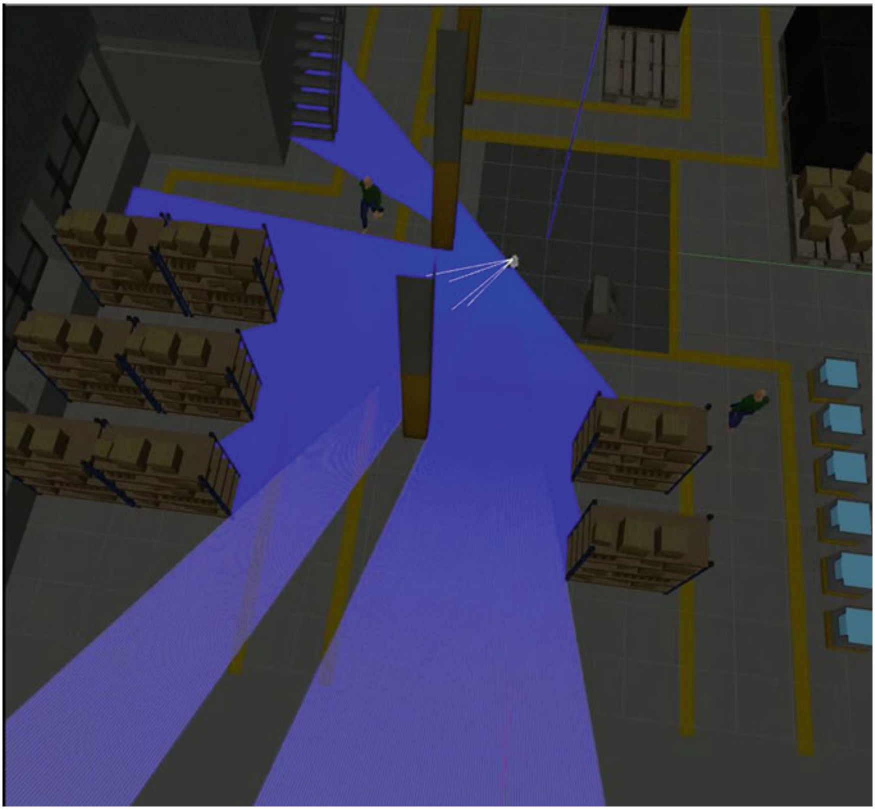


Fig. 3 Robot navigating to the given coordinates

6.1 Faster R-CNN

Faster R-CNN eliminates the use of selective search algorithm as compared to its predecessors R-CNN and fast R-CNN as it uses a separate dedicated network to identify region proposals over the feature map that is generated by convolutional layers. The selective search algorithm was very time-consuming and thus eliminating it resulted in a much faster algorithm. This was developed by [13]. We were getting an accuracy of 91% and an FPS of 4–5 without using a GPU.

6.2 YOLO V4

YOLO V4 was developed in 2020 by [14] and is one of the recently released object detection models. YOLO stands for You Only Look Once, and it is a single-stage

Table 1 Performance of various trained models

Model	Accuracy (%)	FPS
Faster R-CNN	91	4
YOLO V4	94	7
YOLO V4-Tiny	83	12

detector as opposed to faster R-CNN. As the name suggests, a single forward propagation through the neural network is required to detect objects. So, the predictions over an image are calculated by a single algorithm run. Thus, it is faster and more accurate than faster R-CNN. After training the model, it was giving an accuracy of 94% at around 7 FPS.

6.3 YOLO V4-Tiny

YOLO Tiny is a light version of YOLO that was developed to work on mobile devices. It is computationally inexpensive but not as accurate as full-scaled YOLO. After training YOLO V4-Tiny, we were getting an accuracy of 83%, but the frame rate was twice that of both the above models.

7 Results and Discussion

After testing all the trained models, YOLO V4 was chosen as it was able to detect the boxes with maximum accuracy, and the frame rate was not an issue here as the predictions were to be made on a single image that the bot captured after positioning the camera in front of the rack. Table 1 shows the accuracy and speed of trained models.

Figures 4 and 5 show detection of boxes using model trained on YOLO V4 architecture.

Figure 6 shows the output and working of the bot in the environment. It is able to detect and avoid moving obstacles (walking people) as well as fixed obstacles like poles and walls and alter its path accordingly. The left window shows 3D simulation of the robot in Gazebo, while the right window shows 2D map of the environment created by the bot in Rviz, and the camera output is shown in top right window.

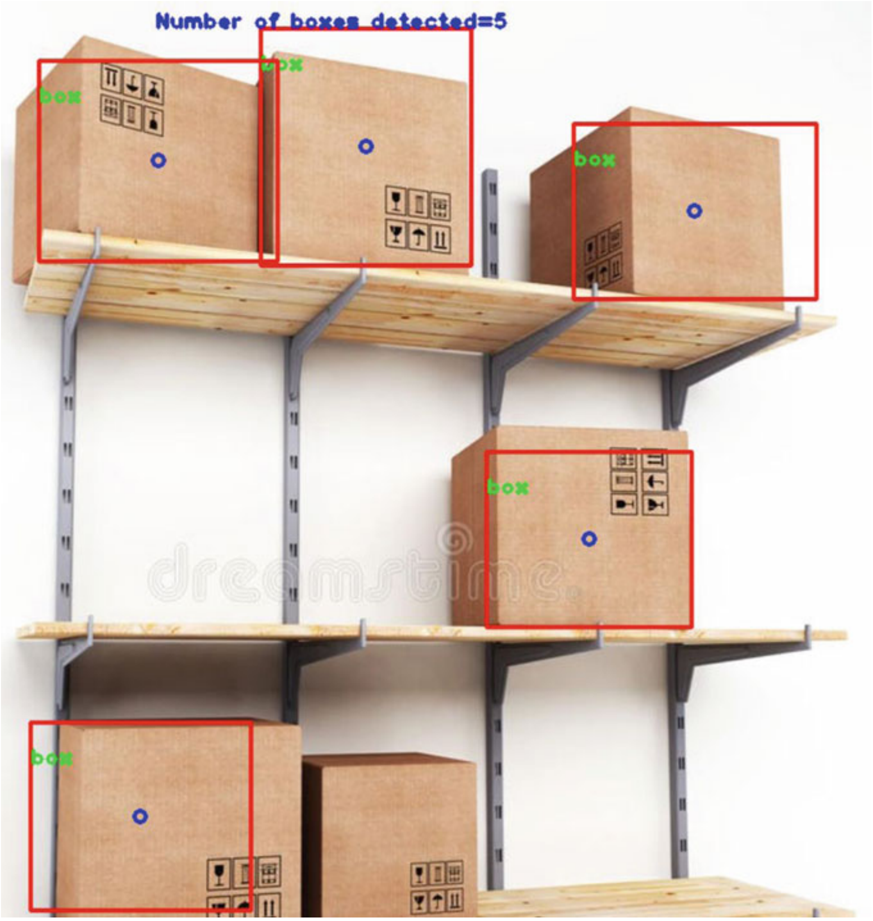


Fig. 4 Detection of boxes using YOLO V4

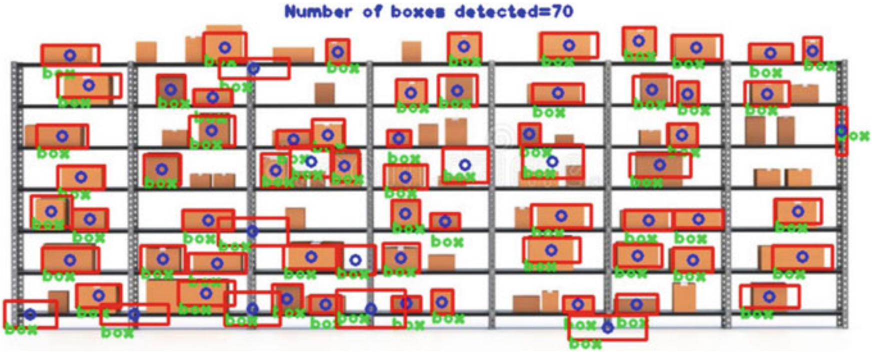


Fig. 5 Detection of boxes using YOLO V4

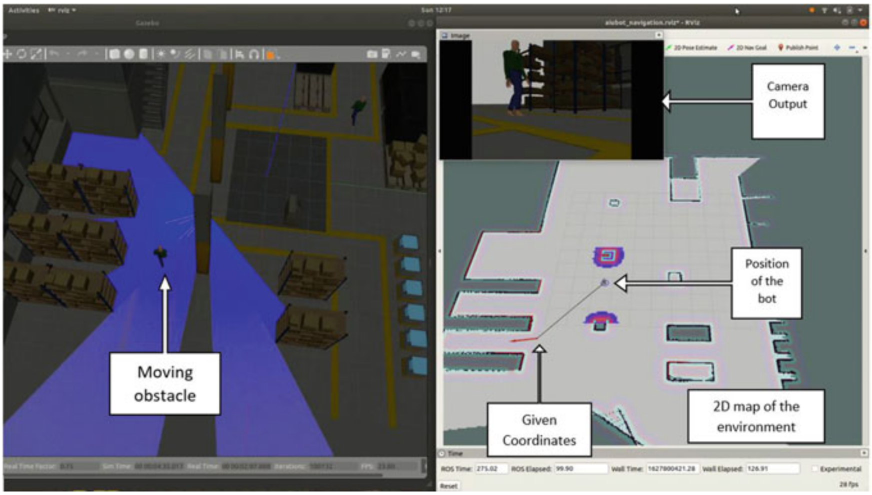


Fig. 6 Gazebo, Rviz and camera output

8 Conclusion

From the results, we can see that the robot can navigate to the coordinates of the warehouse environment given by the operator by avoiding any moving or stationary obstacle, and it is able to detect and count boxes on racks. However, it was observed that for proper detection, the boxes should have some minimum space between each other for more accurate detection. In future, this can be implemented in retail stores, either by using RFID, cameras, or using bar-code scanners mounted on the robot. Such bots can accurately locate each product in store as frequently as we want. This would improve the restocking system, developing money-mapping strategies, and for making better data-driven decisions. It will generate data that aid in requirements and easily integrate with the system.

References

1. Blais F, Rioux M, Domey J (1989) Compact three-dimensional camera for robot and vehicle guidance. Opt Lasers Eng 10(3):227–239
2. Nickerson SB, Jasiobedzki et al (1994) An autonomous mobile robot for known industrial environments. In: Conference on intelligent robotics in field, factory, service, and space, vol 1, pp 12–20
3. Lu F, Milios E (1997) Globally consistent range scan alignment for environment mapping. Autonom Robot 333–349
4. Goel P, Roumeliotis PI, Sukhatme GS (1999) VPH: a new laser radar based obstacle avoidance method for intelligent mobile robots. In: Proceedings of the IEEE international conference on robots and systems, Kyongju, Korea

5. Bakambu JN (2000) Heading-aided odometry and range data integration for positioning of autonomous mining vehicles. In: Proceedings of the IEEE international conference on control applications, Anchorage AK
6. Pfister T, Kriechbaum KL et al (2002) Weighted range sensor matching algorithms for mobile robot displacement estimation. In: IEEE international conference on robotics and automation
7. Ye C, Borenstein J (2002) Characterization of a 2-D laser scanner for mobile robot obstacle negotiation. In: Proceedings of the 2002 IEEE ICRA, pp 2512–2518
8. Sasiadek JZ (2002) Sensor fusion. Annual Rev Control IFAC J 26:203–228
9. Alberto V, Lucas J, Isabel M (2004) Feature extraction and selection for mobile robot navigation in unstructured environments. In: Proceedings of the 5th IFAC symposium on intelligent autonomous vehicles
10. Sasiadek JZ, Polotski et al, Navigation of autonomous mobile robot with gate recognition and crossing. In: Proceedings of the 8th IFAC symposium on robot control (SYROCO'2006), Bologna, Italy
11. Temeltas H, Kayak D (2008) SLAM for robot navigation. EEE Aerosp Electron Syst Maga 23(12):16–19
12. Khairuddin AR, Talib MS, Haron H (2015) Review on simultaneous localization and mapping (SLAM). In: IEEE international conference on control system, computing and engineering (ICCSCE 2015), pp 85–90
13. Ren S, He K, Girshick R, Sun J (2016) Faster R-CNN: towards real-time object detection with region proposal networks. arXiv:1506.01497v3
14. Bochkovskiy A, Chien-Yao W, Mark L (2020) YOLOv4: optimal speed and accuracy of object detection. arXiv:2004.10934v1
15. Tanvir A, Yinglong M, Ahmad B et al (2020) Object detection through modified YOLO neural network. Hindawi Scientific Programming
16. Pal SK, Pramanik A, Maiti J et al (2021) Deep learning in multi-object detection and tracking: state of the art. Appl Intell

Analysis of the Statistical Methods for Vehicle Detection in the Accident Avoidance System an Application of ITS

Diya Vadhwani and Devendra Thakor

Abstract Intelligent Transport Systems consists of applications such as accident warning and avoidance systems. This is an area of interest to us in this research. The accident detection and alert system in the field of ITS (intelligent transportation system) are implemented and analyzed using many statistical methods for proving the accident detection point, GPS Coordinates points search, etc. In this paper, the analysis is performed on how accidents can be prevented using statistical analysis applied to the flow of vehicles in the construction area. This study helps to compare different statistical methods for vehicle detection in accident warning systems. Different statistical methods like Poisson's probability distribution, Binomial distribution and Negative Binomial probability distributions were evaluated considering the count of vehicles and their distances from the area near or far from the prone area considered in the construction site. The aim of this paper is to evaluate and compare the statistical methods on the real-time datasets collected by hardware configuration of Arduino-Mega, Node-MCU and cloud service thingsspeak.com and shows that Poisson's statistical distribution performs better than other methods in terms of probability of success. The different statistics like mean, standard deviation, and variance for different statistical methods are evaluated, out of these Poisson performs better.

Keywords Accident avoidance · Intelligent transportation system (ITS) · Poisson's distribution · Binomial distribution · Negative binomial distribution

1 Introduction

The Intelligent Transportation System (ITS) is the advanced transportation system in which the network is established between roads, vehicles, and users. To improve the security and efficiency of transportation systems, ITS applications provide travelers with information about the road network and communications. ITS provides many application areas to achieve the goals of improving the public safety and human lives.

D. Vadhwani (✉) · D. Thakor
Uka Tarsadia University, Bardoli, India
e-mail: dnvadhwani@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_8

ITS projects are being carried out in big cities such as Delhi, Ahmedabad, Bangalore, Chennai, and Pune in India. Projects underway in India include various functions such as traffic light management, parking lot management, public transport management, and highway toll house maintenance [1]. The accident notification system is implemented using Internet of Things technology. To implement the accident notification application the scenario considered as construction area at highway. Some of the assumptions are used in order to achieve the accuracy in data collection [2].

There are statistical methods like the Poisson's distribution and Binomial distribution and in [3] were analyzed for estimating the probability to classify as hazardous locations.

The Poisson's and Negative-exponential methods were used to find the volume of traffic; the models also show the effect on wildlife when there is more traffic [4].

Number of accidents on roads, traffic flows and traffic density are analyzed depending on count of vehicles that travel at a particular place using Poisson's method [5].

Using negative binomial regression, authors in [6] determined the frequency of accidents at intersections [6] and found an approach that would help reduce the frequency of accidents.

Various statistical methods were used to find the metrics like density of traffic, vehicle count, GPS coordinates of accident, accident frequency, etc. To reduce the crashes at construction sites, the prototype is developed to find count of vehicles that are passing near the prone area here construction site and away from prone area in the construction site. The count of vehicles is the random number considered here for our application In accident warning or notification ITS applications, vehicle detection is one of the key tasks to accomplish. The vehicle detection from the prone area at construction site gives some statistical measures which can be helpful in the safety of public lives.

To find the vehicles which are crossing near the prone area or risky area at construction site was implemented in our work implemented in. Using the Internet of Things, datasets are collected to determine the number of vehicles and their flow rate passing through the risk area. After collecting the dataset, the distance was recorded in a file. Flow analysis of vehicle detection using statistical methods is one of the goals of research to date [7]. In the research study [8], the Poisson method, the negative binomial method, and the binomial method were compared for the first hour. The probability of success which the count of vehicles crossing near the construction site is calculated and shows that probability of success using Poisson statistical distribution method is more as compared to probability of success using negative binomial distribution method and Binomial distribution. Similarly in this paper, the statistical analysis is performed using separately for one-hour, two-hours, and three-hours datasets.

According to author [9], vehicle arrival rates at a given interval point are analyzed using Poisson statistical distribution.

In the statistical evaluation and principle of opportunity, the Poisson statistical distribution is the discrete opportunity distribution, wherein the opportunity of a random range is used to express the occasions which passed off on the constant time interval [9].

In our previous work [7] the prototype was developed to find the real-time datasets. The datasets include distances recorded using random values using ultrasonic sensors configured with Arduino-mega 2560 and Node-MCU Wi-Fi device. After getting the values from the cloud service thingspeak.com, the statistical analysis is performed on the values (here random values). In one hour the number of vehicles is 96 (the random numbers). Each vehicle's distance is recorded from prone area considered and vehicles whose distances were less than 4 m [13] were recorded and considered as there may be an accident occurred. So out of 96, 29 vehicles were the vehicles whose distance was less than 4 m. The distance is considered as 4 m as per rules mentioned in (Road Safety Audits) [13]. Safe distance between vehicles or objects and vehicle is 3 m depending upon speed, the driving conditions, and the type of vehicle used in driving.

Road transportation plays an important role to reduce the use of vehicles by individuals, which directly or indirectly helps to reduce the traffic at roads, air pollution, and use of the combustible source [15].

Intelligent Transport System (ITS) combines various technologies such as collection of real-time data, communication between devices, and machine learning models to provide services in transportation are used to solve the issues in transportation [16]. Some issues like traffic congestion, air pollution, and road accidents have become research area interest, also ITS play important role in reducing these issues [16].

The comparative study between Poisson's, Negative Binomial and Binomial distribution was performed for one-hour time in [8]. The flow rate analysis of vehicles and probability of vehicles crossing near the risky area is evaluated using statistical methods like Poisson distribution, Binomial, and Negative Binomial method. The statistical distribution methods are implemented in python. The probability of success is recorded for each statistical method for one-hour datasets collection.

The main purpose of our research is to provide a comparative analysis of various statistical methods based on the datasets collected to record the vehicle distance from the risk area of the construction site. The distances of vehicle were recorded for 1 h, 2 h, and 3 h. To find the flow rate analysis and count of vehicles the different statistical methods were implemented in python.

Apart from probability standard deviation, variance and mean was recorded for each statistical method. Because the analysis is a one-hour data collection, use statistical analysis of vehicle detection in an accident warning or notification system to get good results regarding the probability of success. Hence computations for two-hour and three hours datasets were evaluated to find the probability of success and different statistics.

2 Objectives and Precautionary Measures

2.1 Objectives of Study

Vehicle detection in accident warning and avoidance systems is an important task. Since this research aims to avoid accidents at construction sites, we will consider a scenario in which a prototype is implemented in the previous work.

Following are the objectives for this research study:

- Real-time data collection (i.e., counts of vehicles per unit time).
- Flow rate analysis of vehicles per hour.
- Statistical Analysis of Vehicle Detection using Poisson's probability distribution, Negative Binomial Distribution, and Binomial probability distribution.
- To find the probability of success of vehicle arrivals with Statistical methods like Binomial distribution, Poisson's distribution, and Negative Binomial Distribution.
- Finding statistics like mean, standard deviation, and variance for different statistical methods.
- Hypothesis test for Poisson's Probability distribution.

Main Contribution of this research paper:

- To develop the algorithms for Poisson distribution, Binomial distribution, and Negative Binomial distribution methods.
- Collect 1 h, 2 h, and 3 h records in real-time from thingspeak.com.
- To compare the probability of success of vehicle arrivals using different statistical methods like Poisson, Binomial and Negative Binomial.
- Calculating the statistics like mean, standard deviation, and variance for different statistical methods.
- To perform hypothesis test for Poisson's distribution.

2.2 Precautionary Measures

Road accidents lead to fatal deaths for human beings, this research study aims to reduce the accidents at construction sites and save the human being.

Following are some of the Precautionary measures to be considered while implementing the scenarios for accident prevention at construction sites [14].

- Vehicles should be with airbags installed in it along with maintained breaks and good lighting.
- Awareness for traffic rules for driver.
- Installation of necessary modules in vehicles while crossing near construction area.
- Proper use of helmet by driver of vehicle.

3 Proposed System

In the proposed system the three statistical methods like Poisson's distribution, Binomial and Negative Binomial are used for evaluation of probability of success for vehicle arrival to find the accident occurrences in the construction area. This section explains the different algorithms for comparative analysis of different statistical methods like Poisson's method, Binomial method, and Negative Binomial.

3.1 Poisson Probability Distribution Method

Poisson's method is used in the flow rate analysis of vehicles. According to book Chap. 4 from [10], If n is the Poisson random Variable than the Poisson Probability Mass Function (PMF) is given by following equation [7, 8].

$$P(n) = \lambda\hat{}\, n\, e\hat{} - \lambda/n!. \tag{1}$$

Following are the notation used in algorithm evaluation [7, 8] (Table 1).

The Poisson probability distribution algorithm is shown below to determine the probability of vehicle arrival in 1 h, 2 h, and 3 h.

```
-----------------------------------------------------------
Algorithm 1: Vehicle Detection using Poisson's Probability distribution
-----------------------------------------------------------
1. (Input: Distance of Vehicles)
2. (Output: Probability of success of vehicle
3. Define the Poisson's Probability Distribution Method
4. Find the distance 'd'(random value)of vehicle from prone area;
5. Set the threshold=4.
6. if distance < threshold then
7. count the number of vehicles which crosses near the risky area.
8. Send the warning to workers using LED or Buzzer
```

Table 1 Notations used for Poisson's distribution

Notations for Poisson's distribution	Meaning/descriptions
P (n) = λ ^ n e^ − λ/n!	Probability of exactly 'n' vehicles which arrived in given time using Poisson's Probability distribution Method
λ	Average arrival rate of vehicles
P (n)	Is the probability of exactly 'n' vehicles arrived in the given time interval
N	The number of vehicles arriving at a particular time interval
E	E = 2.71828 which is constant value for natural logarithm base system)

```
9. else
10. count the number of vehicles which are far from the risky area.
11. Do not send the warning to workers using LED or Buzzer.
12. endif
13. Find the vehicle less than and more than the threshold value(here
'n').
14. Find the value of 'λ' (For 1hour,2hours, 3hours)
       λ=(I/3600)t (Average number of vehicles(I)passing through
the particular time interval t(s))
      where, I1=96 for λ1, t=60
         I2=193 for λ2, t=60
         I3=310 for λ3, t=60
15. e=2.711828 ('e' is the constant value for natural logarithm
system).
16. Calculate the probability of random value here, vehicle arrival
using Poisson's Probability Distribution method as:
              P (n) = λ^n e^ − λ / n!
----------------------------------------------------------------
```

3.2 Binomial Distribution Method

The binomial probability distribution method is used for vehicle flow analysis.

Probability Mass function for Binomial probability distribution is given by [7, 8]:

$$b(x; n, P) = nCx * P^{\wedge}x * (1 - P)^{\wedge}n. \tag{2}$$

Following is the notation used in algorithm evaluation (Table 2).

The algorithm for statistical binomial's probability distribution is stated below to calculate the probability of vehicles arrived in 1 h, 2 h, and 3 h.

```
----------------------------------------------------------------
Algorithm 2: Vehicle Detection using Binomial Probability Distri-
bution
----------------------------------------------------------------
1. (Input: Distance of Vehicles)
2. (Output: Probability of success of vehicle
```

Table 2 Notations used for binomial distribution

Notations for Binomial distribution	Meaning/descriptions
b(x; n, P) = nCx * P^x * (1 − P) ^n − x	Binomial Probability distribution for vehicle arrivals
b	Binomial Statistical Probability distribution
X	The total number of "successes" (Here, the number of vehicles is counted in or near the danger zone)
P	Probability of success for a vehicle arrival
N	Total number of Vehicles

```
3.Define the Binomial Distribution Method
4. Find the distance 'd'(random value)of vehicle from prone area;
5. Set the threshold=4
6. if distance < threshold then
7. count the number of vehicles that crosses near the restriction area.
8. Send the warning to workers using LED or Buzzer
9. else
10. count the number of vehicles that crossed far from risky area.
11. Do not send the warning to workers using LED or Buzzer.
12. endif
13. For each hour calculate the probability distribution using Binomial Probability Method
14. Calculate the probability of random value here vehicle arrivals using Statistical Binomial Probability Distribution for 1hour
            $b(x; n, P) = nCx * Px * (1 - P)n - x$
Where: b = Binomial probability distribution, x = The total number of successes (Here, Vehicles that crossed near risky or danger area)
x=29
P=0.3
15. Calculate the probability of random value here vehicle arrival using Binomial Probability Distribution for 2hours
            $b(x; n, P) = nCx * Px * (1 - P)n - x$
Where: b = Binomial probability distribution, x = Total number of "successes" (Here, Vehicles crossed near risky area)
x=53
P=0.274
16. Calculate the probability of random value here vehicle arrival using Binomial Probability Distribution for 3hours
            $b(x; n, P) = nCx * Px * (1 - P)n - x$
Where: b = Binomial probability distribution, x = Total number of "successes" (Here, Vehicles crossed near risky area).
x=80
P=0.25
----------------------------------------------------------------
```

3.3 *Negative Binomial Distribution Method*

Negative Binomial method is used in the flow rate analysis of vehicles.

Probability Mass function for Negative Binomial distribution is given by [7, 8]:

$$b * (x; r,P) = x - 1Cr - 1 * P^{r} * (1 - P)^{\wedge}x - r. \tag{3}$$

Following are the notation used in algorithm evaluation (Table 3).

The algorithm for Negative Binomial's distribution is stated below to evaluate the probability of vehicles arrived at 1, 2 and 3 h.

Table 3 Notations used for negative binomial distribution

Notations for negative binomial distribution	Meaning/descriptions
b*(x; r, P) = x-1Cr-1 * P^r * (1 − P) ^x − r	Negative binomial probability for vehicles arrived
X	X is the number of vehicles needed to find r successes
R	r successes in a negative binomial method
P	P is the probability (here successful) that the vehicle arrived using the negative binomial method

```
---------------------------------------------------------------
Algorithm 3: Vehicle Detection using Negative Binomial Distribution
---------------------------------------------------------------
1. (Input: Distance of Vehicles)
2. (Output: Probability of success of vehicle
3. Define the Negative Binomial Distribution Method
4. Find the distance 'd'(random value)of vehicle from risky area;
5. Set the threshold=4
6. if distance < threshold then
7. count the number of vehicles that crossed near the risky area.
8. Send the warning to workers using LED or Buzzer
9. else
10. count the number of vehicles that crosses far the risky area.
11. Do not send the warning to workers using LED or Buzzer.
12. endif
13. For each hour calculate the probability distribution using Negative Binomial Method
14. Calculate the probability of random value here vehicle arrival
using Negative Binomial Probability Distribution for 1hour
    b*(x; r, P) = x-1Cr-1 * Pr * (1 - P)x - r,
Where x =96= vehicles per hour crossed.
r=29 vehicles having distance less than threshold
(i.e., number of success)
P=0.3 is the probability of vehicle arrived, here success.
15. Calculate the probability of random value here vehicle arrival
using Negative Binomial Probability Distribution for 2hours
    b*(x; r, P) = x-1Cr-1 * Pr * (1 - P)x - r,
Where x =193= vehicles per hour crossed.
r=53 vehicles having distance less than threshold
(i.e., number of success)
P=0.274 is a probability of success
16. Calculate a probability of random value here, vehicle arrival
using Negative Binomial Probability Distribution for 3hours
    b*(x; r, P) = x-1Cr-1 * Pr * (1 - P)x - r,
Where x =310= vehicles per hour crossed.
r=80 vehicles having distance less than threshold
(i.e., number of success)
P=0.25 is the probability of success
---------------------------------------------------------------
```

4 Experimental Setup and Results

With new technologies such as the Internet of Things, intelligent transportation systems help improve people's safety. In this research study, vehicle detection and statistical analysis of mobile vehicles in an accident warning or notification and avoidance system was implemented using the Arduino Mega2560 microcontroller and Wifi Node-MCU device [7, 8]. The distance of a moving vehicle from the danger area is evaluated and flow rate of vehicles is analyzed.

This research paper aims to compare the different the statistical methods evaluated for 1 h, 2 h and 3 h real-time data collected (here random values collected using prototype developed in our research study [7, 8]).

Compare various statistical distribution methods such as Poisson distribution, binomial distribution, and negative binomial probability distribution to determine the probability success of vehicle arrival in 1 h, 2 h, and 3 h data collection (Fig. 1).

After the implementation of prototype the datasets were collected for 1 h, 2 h, and 3 h to find the different statistics like vehicle flow rate and total count of the vehicles. Following Fig. 2 shows the distances calculated using sensors as.

The graph for vehicles crossed in 1 h, 2 h, and 3 h for Poisson's Probability distribution, Binomial Probability distribution, and Negative Binomial probability distribution methods are shown below (Figs. 3, 4, 5, 6, 7, 8, 9, 10 and 11).

To determine the number of vehicles crossing the danger zone or near the hazard zone, the distance is assumed to be 4 m [13] to 400 m. If the distance from vehicle to vehicle is less than 4 m, there is a possibility of an accident. Accidents may occur or be reduced if the distance from vehicle exceeds 4 m [7, 8].

The real-time dataset provides values (these are random values) for 1 h, 2 h, and 3 h distances.

From the dataset [7, 8]:

Total number of vehicles arriving in 1 h = 96 (random value here).

Total number of vehicles arriving within 2 h = 193 (random value here).

Total number of vehicles arriving within 3 h = 310 (random value here).

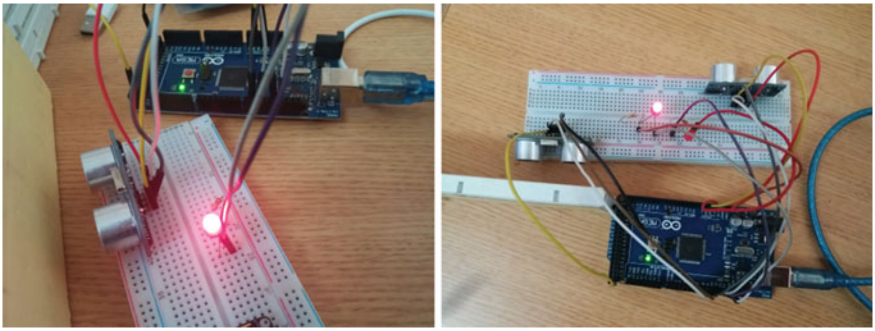
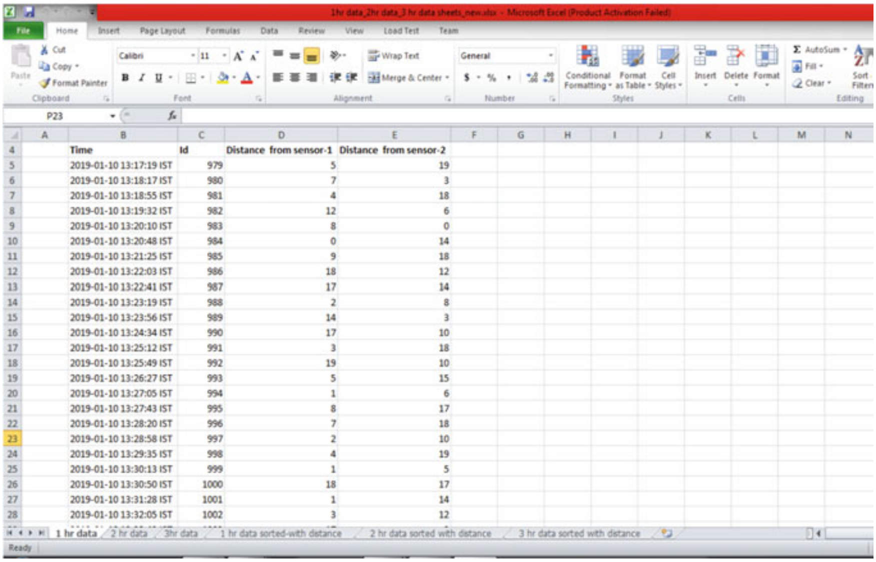


Fig. 1 Implementation of prototype for distances less than 4 m and more than 4 m [7, 8]



	A	B	C	D	E
4		Time	Id	Distance from sensor-1	Distance from sensor-2
5		2019-01-10 13:17:19 IST	979	5	19
6		2019-01-10 13:18:17 IST	980	7	3
7		2019-01-10 13:18:55 IST	981	4	18
8		2019-01-10 13:19:32 IST	982	12	6
9		2019-01-10 13:20:10 IST	983	8	0
10		2019-01-10 13:20:48 IST	984	0	14
11		2019-01-10 13:21:25 IST	985	9	18
12		2019-01-10 13:22:03 IST	986	18	12
13		2019-01-10 13:22:41 IST	987	17	14
14		2019-01-10 13:23:19 IST	988	2	8
15		2019-01-10 13:23:56 IST	989	14	3
16		2019-01-10 13:24:34 IST	990	17	10
17		2019-01-10 13:25:12 IST	991	3	18
18		2019-01-10 13:25:49 IST	992	19	10
19		2019-01-10 13:26:27 IST	993	5	15
20		2019-01-10 13:27:05 IST	994	1	6
21		2019-01-10 13:27:43 IST	995	8	17
22		2019-01-10 13:28:20 IST	996	7	18
23		2019-01-10 13:28:58 IST	997	2	10
24		2019-01-10 13:29:35 IST	998	4	19
25		2019-01-10 13:30:13 IST	999	1	5
26		2019-01-10 13:30:50 IST	1000	18	17
27		2019-01-10 13:31:28 IST	1001	1	14
28		2019-01-10 13:32:05 IST	1002	3	12

Fig. 2 Real-time dataset having distance value of vehicle

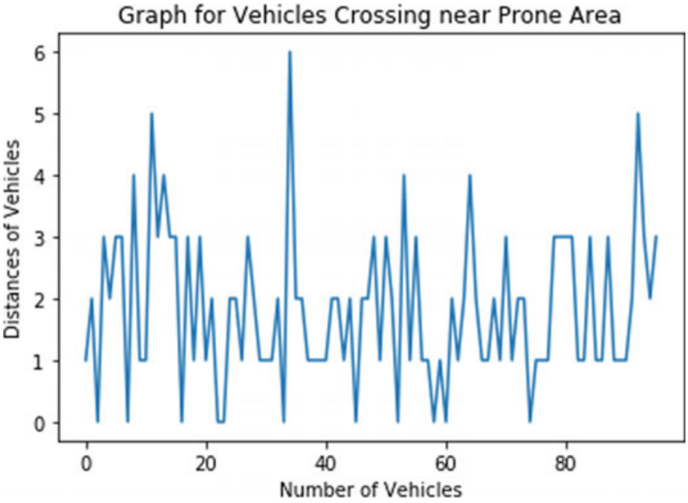


Fig. 3 Count of vehicles with their distances from risky areas using Poisson's probability distribution in 1 h

Random values for 1 h, 2 h, and 3 h vehicle distances were considered to determine the probability of success using various statistical methods. We evaluated the probability of success and different statistics and performed a comparative analysis between different statistical methods.

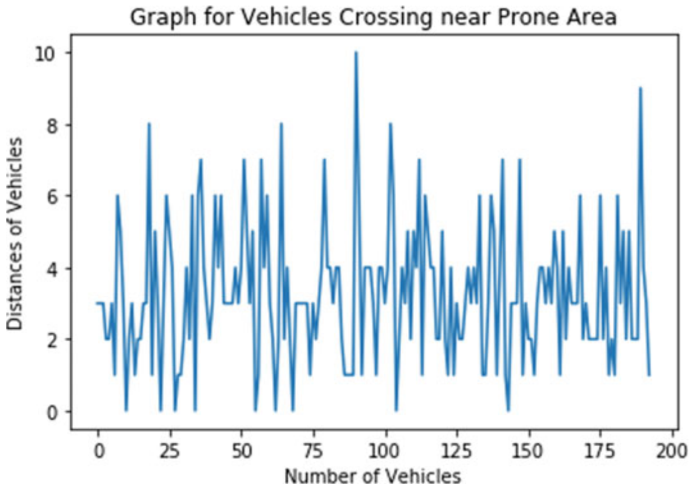


Fig. 4 Count of vehicles with their distances from risky areas using Poisson's probability distribution in 2 h

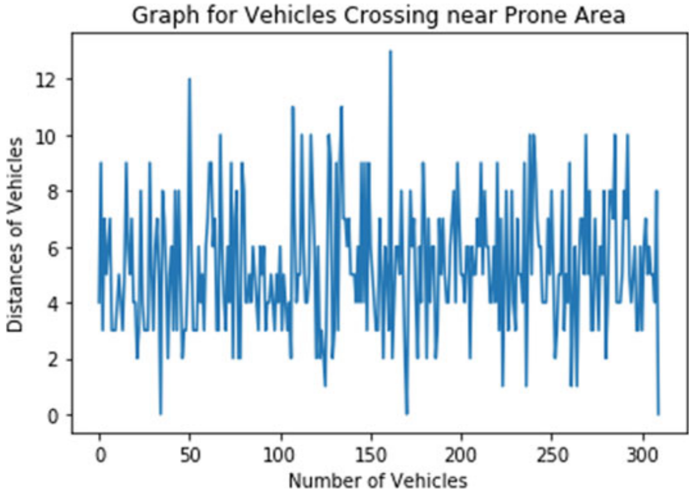


Fig. 5 Count of vehicles with their distances from risky areas using Poisson's probability distribution in 3 h

4.1 Results for All Three Statistical Distributions for 1 h Statistic

See Tables 4 and 5 and Fig. 12.

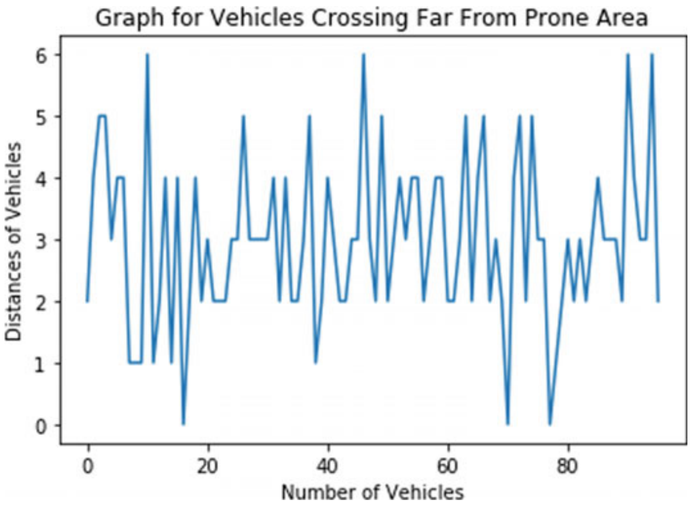


Fig. 6 Count of vehicles with their distances from risky areas using binomial probability distribution in 1 h

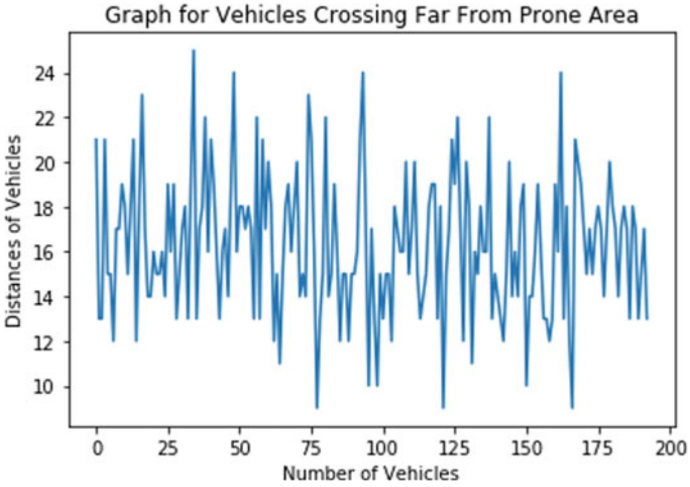


Fig. 7 Count of vehicles with their distances from risky areas using binomial probability distribution in 2 h

4.2 Results for All Three Statistical Distributions for 2 h Statistic

See Table 6 and 7 and Fig. 13.

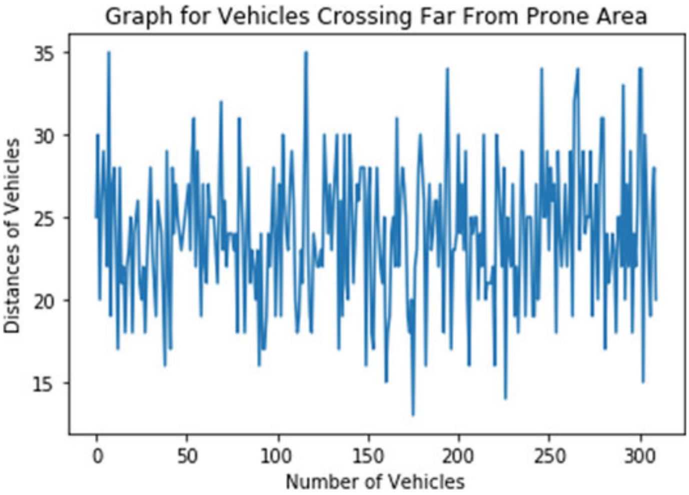


Fig. 8 Count of vehicles with their distances from risky areas using binomial probability distribution in 3 h

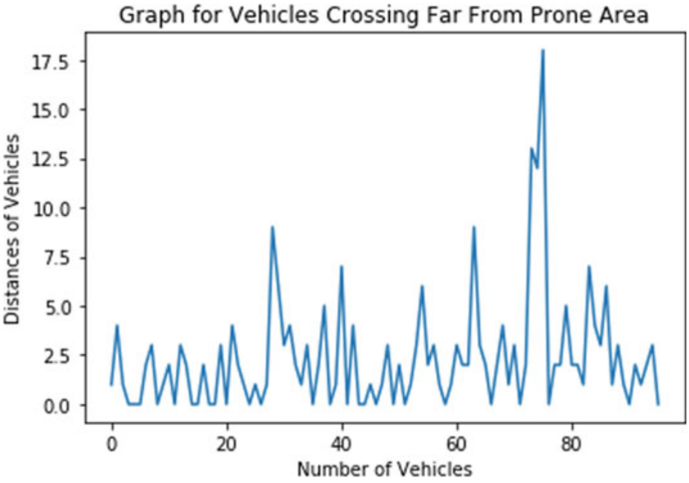


Fig. 9 Count of vehicles with their distances from risky areas using negative binomial probability distribution in 3 h

4.3 Results for All Three Statistical Distributions for 3 h Statistic

See Tables 8 and 9 and Fig. 14.

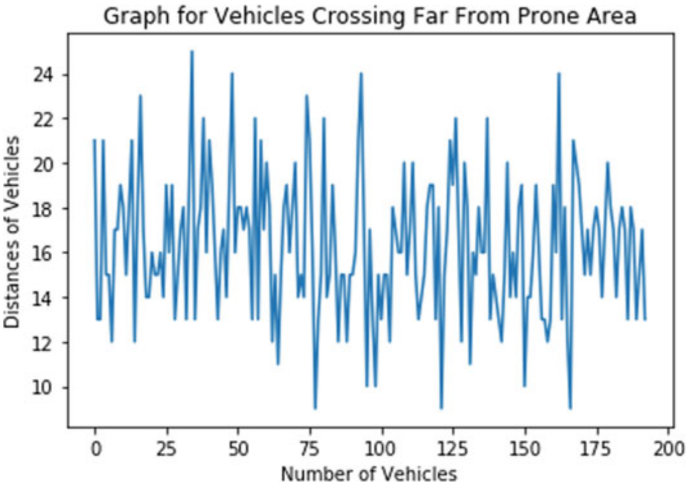


Fig. 10 Count of vehicles with their distances from risky areas using negative binomial probability distribution in 2 h

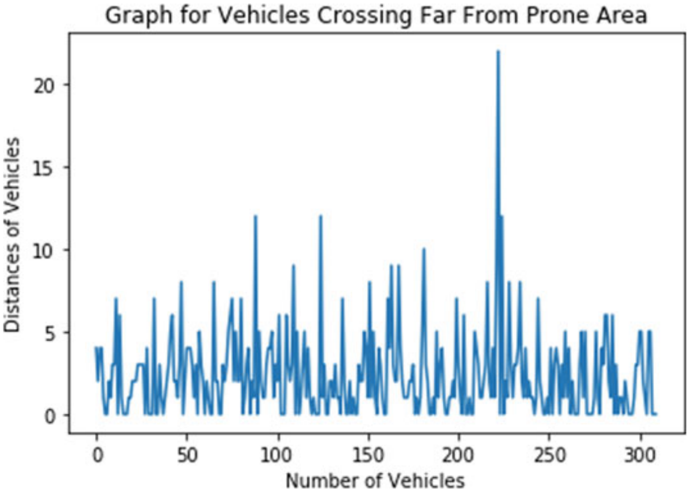


Fig. 11 Count of Vehicles with their distances from risky areas using negative binomial distribution in 3 h

Table 4 Comparative study of 1 h vehicle arrival probability

Statistical methods	P(X = x)	P(X <= x)	P(X >= x)
Poisson's distribution	0.3230	0.52489	0.7981
Binomial distribution	0.0883	0.56761	0.52069
Negative binomial distribution	0.00461	0.96318	0.04143

Table 5 Comparative study of various statistics for 1 h

Statistical methods	Mean	Standard deviation	Variance
Poisson’s distribution	1.6	1.6	1.264
Binomial distribution	28.8	4.49	20.16
Negative binomial distribution	67.667	15.019	225.556

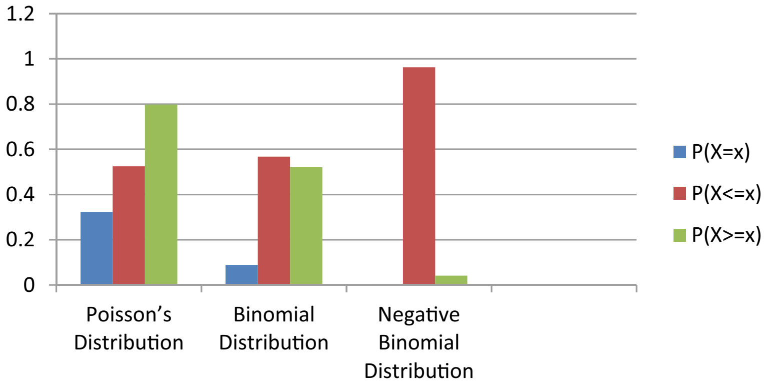


Fig. 12 Comparative analysis of the probabilities of a 1 h statistical method

Table 6 A comparative study of 2 h vehicle arrival probabilities

Statistical methods	P(X = x)	P(X <= x)	P(X >= x)
Poisson’s Distribution	0.130	0.170	0.9596
Binomial distribution	0.063	0.593	0.469
Negative binomial Distribution	0.014	0.684	0.330

Table 7 A two-hour comparative study of various statistics

Statistical methods	Mean	Standard deviation	Variance
Poisson’s distribution	3.21	1.79	3.21
Binomial distribution	52.11	6.168	52.11
Negative binomial distribution	143.2	23.08	530.72

From results, it is observed that for 1 h dataset, the mean, standard deviation, and variance for Poisson’s distribution is 1.6, 1.6, and 1.264, similarly for two hours, it is 3.21, 1.79, and 3.21, for three is 5.16, 2.72 and 5.16.

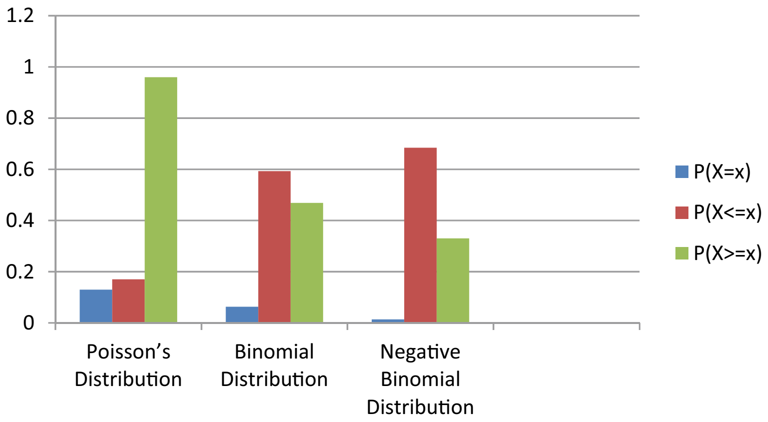


Fig. 13 Comparative analysis of the probabilities of a two-hour statistical method

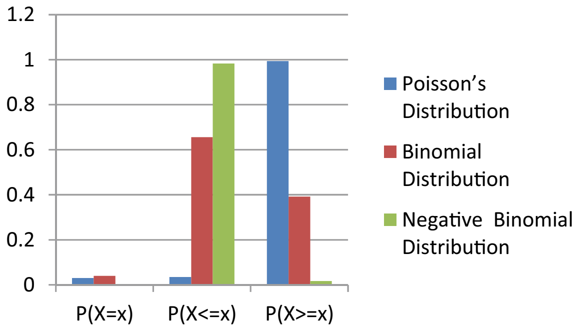
Table 8 A comparative study of three-hour vehicle arrival probabilities

Statistical methods	P(X = x)	P(X <= x)	P(X >= x)
Poisson's distribution	0.030	0.035	0.994
Binomial distribution	0.040	0.656	0.392
Negative binomial distribution	0.00115	0.983	0.0172

Table 9 A three-hour comparative study of various statistics

Statistical methods	Mean	Standard deviation	Variance
Poisson's distribution	5.16	2.72	5.16
Binomial distribution	77.5	7.624	58.125
Negative binomial distribution	240	30.98	960

Fig. 14 Comparative analysis of the probabilities of a 3 h statistical method



5 Hypothesis Test

Hypothesis test for Poisson distribution is very similar to binomial distribution. Let $\alpha = 0.05$ be the level of significance, If the probability of success is greater than α, then the null hypothesis is to be accepted. And if probability is less than α, then the alternative hypothesis is accepted [11].

Let n = Number of Occurrences, P(n) = Exact Probability, p(x < = n) and p(x > = n) are cumulative probabilities, F(n) is computed frequency.

Let p (n) be the probability that a large number of vehicles will cross near a dangerous or vulnerable area at a distance of less than 4 m.

Total vehicle arrived in 1 h = 96, Vehicles with distance less than 4 m = 29, Probability of success = 0.323.

Let Null hypothesis: H0 = "Vehicle count with distances less than 4 m" p(n) = 0.323.

Alternative hypothesis: HA = "Vehicle count with distances more than 4 m".

Test statistic: X = number of vehicles in 1 h crossing near risky or prone area.

Null distribution: This Probability function is based on the null hypothesis.

Let the Rejection region: for the null hypothesis, it is expected to get 29 vehicles crossing near prone area whose distance is less than 4 m in one hour. So we will reject H0 if the number of vehicles crossed far from risky or prone areas whose distance is more than 4 m. The rejection region as {5, 6, 7, 8, 9, 10}. That is, if there are a number of vehicles in this area that are less than 4 m in an hour, we reject the hypothesis that the vehicle is crossing far away from the hypothetical danger zone of n.

Let us see the probability table for the null distribution.

The Cumulative probabilities of Poisson's distribution for 1 h were calculated in previous study [7, 8] as follows (Tables 10, 11 and 12).

Table 10 Cumulative probabilities of Poisson' distribution for 1 h [7, 8]

N	p(n)	p(x <= n)	p(x >= n)	F(n)
0	0.20189	0.20189	1	12.11
1	0.323	0.5248	0.7981	19.38
2	0.2584	0.78729	0.47507	15.509
3	0.1378	0.92112	0.21664	8.2698
4	0.0551	0.97631	0.07881	3.314
5	0.01764	0.99392	0.02368	1.0584
6	0.004704	0.998624	0.00604	0.28226
7	0.00107	0.999624	0.00134	0.64512
8	0.00215	0.999692	0.00026	0.0129
9	0.00038	0.9999	0.00005	0.00229
10	0.00001	0.999994	0.000001	0.00001
11	0	1	0	

Table 11 Cumulative probabilities of Poisson' distribution for 2 h

N	p(n)	p(x <= n)	p(x >= n)
0	0.0403	0.0403	1
1	0.13	0.17	0.9596
2	0.208	0.378	0.8301
3	0.222	0.6002	0.62218
4	0.179	0.779	0.39971
5	0.115	0.893	0.22117
6	0.0613	0.995	0.10655
7	0.028	0.983	0.04523
8	0.11	0.994	0.1711
9	0.004	0.998	0.00583
10	0.0012	0.999	0.0018
11	0.0003	1	0.00003

Table 12 Cumulative probabilities of Poisson' distribution for 3 h

N	p(n)	p(x <= n)	p(x >= n)
0	0.006	0.006	1
1	0.03	0.035	0.99426
2	0.076	0.112	0.96463
3	0.131	0.243	0.8819
4	0.17	0.413	0.7567
5	0.175	0.588	0.5871
6	0.151	0.738	0.41209
7	0.111	0.849	0.26157
8	0.072	0.921	0.15061
9	0.041	0.962	0.079
10	0.021	0.983	0.03801
11	0.01	0.997	0.0168

Similar to one-hour computations for hypothesis the rejection regions for two-hours cumulative probabilities is {0, 7, 8, 9, 10, 11}.

Similar to one-hour and two-hours computations for hypothesis the rejection regions for three hours cumulative probabilities is {0, 1, 9, 10, 11}.

6 Conclusion and Future Scope

The Poisson's probability distribution, Binomial probability distribution, and Negative Binomial probability distributions are implemented to calculate and count of vehicles and vehicles passing near risk or restricted areas.

A comparative study of statistical methods has shown that the Poisson method has a success probability of 0.3 for an hour of vehicle arrival.

Similarly for 2 h and 3 h were calculated, in which all three statistical methods give different probability of vehicles arrivals. From 2 h statistics, the Poisson method has probability of 0.13 and Binomial has 0.063, while Negative Binomial Method has 0.0149. From 3 h statistics, the Poisson method has probability of 0.30 and Binomial has 0.04, while Negative Binomial Method has 0.00115. It can be seen that when the time is increased in terms of evaluating the flow rate of vehicles the probability of success of vehicle arrival decreases.

The different statistics like mean, standard deviation, and variance is also evaluated for Poisson's, Binomial, and Negative Binomial Method.

These statistics show that the Poisson methods have high probabilities on the occurrence of vehicles as compared to binomial and negative binomial. So the Poisson's method gives better results in terms of finding the probabilities of success for an event (here vehicle), hence Poisson distribution is the method is be used to evaluate the accident ratio at construction site.

Finally, the probabilities of success using Poisson's method show some measures which can be helpful to avoid the accidents. So after comparison of different statistical methods, it is seen that Poisson's method is used to avoid accidents in accident notification systems an application of intelligent transportation systems.

The Poisson distribution hypothesis test shows a rejection zone where the probability is lower than the significance level. The hypothesis test, in an hour's calculation, shows the null hypothesis accepted in the rejected region {5, 6, 7, 8, 9, 10}. Similarly to, the rejection ranges for the 2 h and 3 h calculations are {0, 7, 8, 9, 10, 11} and {0, 1, 9, 10, 11}. As a part of future the traffic density parameter, and evaluation of more number statistical distribution methods for same dataset will be performed and finally will be compared with the existing methods.

References

1. Rawal, Tejas, and V. Devadas.: Intelligent transportation system in india-a review. Institute of Development Management (2015): 299.
2. Vadhwani, Diya Naresh, and Sanjay Buch.: A Novel Approach for the ITS Application to Prevent Accidents using Wireless Sensor Network, IoT and VANET. 2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT). IEEE, 2019.
3. Al-Ghamdi, A. S.: Probability approach for ranking high-accident locations. WIT Transactions on The Built Environment 49 (2000).
4. Martolos J, Andel P (2013) Distances between Vehicles in traffic Flow and the Probability of Collision with Animals. Transactions on Transport Sciences 6(2):97

5. Idrissa, Kayijuka.: Mathematical Study for Traffic Flow and Traffic Density in Kigali Roads. International Journal of Mathematical and Computational Sciences 11.3 (2017): 104–108
6. Poch, Mark, and Fred Mannering.: Negative binomial analysis of intersection-accident frequencies. Journal of transportation engineering 122.2 (1996): 105–113.
7. Vadhwani, D., & Thakor, D. (2021). Statistical analysis of vehicle detection in the ITS application for monitoring the traffic and road accident using internet of things. In *Advances in VLSI and Embedded Systems* (pp. 55–70). Springer, Singapore.
8. Vadhwani, D., & Thakor, D. (2021). Comparative analysis of statistical methods for vehicle detection in the application of ITS for monitoring traffic and road accidents using IoT. In *Data Science and Intelligent Applications* (pp. 355–361). Springer, Singapore.
9. Gerlough, D. L., & Schuhl, A. (1955). *Use of Poisson distribution in highway traffic* (pp. 1–58). Saugatuck, Conn.: Eno Foundation for Highway Traffic Control.
10. Haight FA (1967) Handbook of the Poisson Distribution. John Wiley & Sons, New York
11. Kissell, R., & Poserina, J. (2017). Advanced Math and Statistics. *Optimal Sports Math, Statistics, and Fantasy*, 103–135.
12. Orloff, J., & Bloom, J. Null Hypothesis Significance Testing I Class 17, 18.05.
13. Road Safety Audits (Information for safe following distances between vehicles)[Online]. https://www.qld.gov.au/transport/safety/rules/road/distances [Accessed on 20 August 2021].
14. Gopalakrishnan S (2012) A public health perspective of road traffic accidents. Journal of family medicine and primary care 1(2):144
15. Patel, D., Narmawala, Z., Tanwar, S., & Singh, P. K. (2019). A systematic review on scheduling public transport using IoT as tool. *Smart innovations in communication and computational sciences*, 39–48.
16. Zear, A., Singh, P. K., & Singh, Y. (2016). Intelligent transport system: A progressive review.
17. Ye X, Wang K, Zou Y, Lord D (2018) A semi-nonparametric Poisson regression model for analyzing motor vehicle crash data. PLoS ONE 13(5):e0197338
18. Nugra H, Abad A, Fuertes W, Galarraga F, Aules H, Villacis C, Toulkeridis T (2016, September). A low-cost IoT application for the urban traffic of vehicles, based on wireless sensors using GSM technology. In *2016 IEEE/ACM 20th International Symposium on Distributed Simulation and Real Time Applications (DS-RT)* (pp. 161–169). IEEE.

An Algorithm to Detect Malicious Activity in Dynamic Vanet Environment

Gagan Preet Kour Marwah and Anuj Jain

Abstract The technology of vehicle ad hoc networks (VANETs) evolved from that of mobile ad hoc networks (MANETs), which help to improve the performance of the transportation area. Because of its benefits in assuring traffic safety and preventing accidents, this technology is gaining a lot of traction. Furthermore, a VANET exhibits self-organization, fast topological changes, and frequent link disconnection, all of which might pose problems. A very effective technique is necessary to mitigate these concerns; hence, this work has used a firefly optimization algorithm (FOA) and a whale optimization algorithm (WOA) as a hybrid model. As a result, the developed model is known as HFWOA-VANET, and it combines the benefits of both metaheuristic methods and is used to improve VANET performance. This procedure is primarily based on the analysis of each vehicle's Quality of Service (QoS) criteria. As a result, the vehicle's performance may be determined, allowing for better service under the VANET platform. This study is implemented on the NS2 platform, and the results are examined to ensure that the suggested model performs as expected. Furthermore, the model's performance is compared to that of existing technology; as a result, the proposed model can be assured to be the most effective technique in terms of performance metrics.

Keywords VANET · Cluster · FOA · WOA · QOS · QMM-VANET · HFWOA-VANET

1 Introduction

Road safety issues have become increasingly prevalent in the recent years, prompting researchers to seek solutions, as road accidents are one of the leading causes of death. According to the World Health Organization (WHO), road accidents are the eighth largest cause of death, with the possibility of being the fifth leading cause of

G. P. K. Marwah (✉) · A. Jain
School of Electronics and Electrical Engineering, Lovely Professional University, Phagwara, Punjab, India
e-mail: gaganmarwaha.marwaha@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_9

death by 2030 [3]. As a result, vehicular ad hoc networks (VANETs) are an excellent solution for overcoming traffic concerns like as accidents, impediment risk, halting problems, and financial overhead [5] since it allows vehicles to interact with one another. VANETs are a subclass of mobile ad hoc networks (MANETs) [9] that require each vehicle to deliver messages to all other vehicles using road side units (RSUs) or multi-hop communication [1]. In contrast to MANETs, VANETs are self-organizing and dynamic mobile networks based on moveable vehicles and road communication infrastructure. In general, it is used in a variety of situations, including sending messages in an emergency, ensuring vehicle safety, and providing mobile entertainment [2]. Furthermore, using vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) connections via the on board unit (OBU), and road side unit (v) devices, it provides several safety and non-safety assistances to road users [4, 10, 11]. In road traffic, VANETs are used as a component to increase road safety by informing drivers of any accidents that have happened or offering online access to travelers [6]. Typically, GPS and sensors are used in vehicles to monitor the car's state, such as device performance, geographic representation of vehicle position, and road conditions. Even yet, there are several challenges with VANET, such as topology creation and repetitive link separation. Because data is carried via a multi-hop wireless network, improving VANET reliability becomes a difficult task [2]. As a result, routing protocols are used in the VANET. In VANET, there are reactive, proactive, hybrid, and geographic routing protocols that are based on topology, location, geocast, broadcast, and cluster. The cluster-based routing protocol is the most efficient of these routing protocols since it uses less energy among the sensor nodes. The cluster-based routing protocol (hybrid routing protocol) combines the reactive and proactive routing protocols' properties, and these protocols are classified as topology-based routing protocols [12]. Clusters can also be characterized as nodes in a hybrid network that are combined in a specific portion. Cluster-based protocols are designed to increase network scalability by allowing nodes to interface with clusters using proactive routing protocols controlled by selected cluster heads. Reactive routing protocols are engaged if there is interaction between clusters [7]. Furthermore, the protocol's effectiveness is measured using Quality of Service (QoS) parameters such as latency, delay, packet delivery ratio, throughput, and so on [8]. Whereas, proactive and reactive protocols achieve QoS targets because they have the potential to pick alternate routes in advance, and even in the event of a link loss, they can reduce the time necessary for convergence [13]. Because QoS parameters can reduce the routing protocol's performance, optimization is required to tune the constants and parameters of the timers to maximize the protocol's effectiveness in terms of accuracy. Hybrid optimization algorithms are incorporated with the cluster-based routing protocol because optimization allows routing protocols to resolve multi-objective issues while also increasing the protocol's computation speed.

The key contributions of this paper are summarized in this article as:

- Detection of malicious activity by making a hybrid metaheuristic algorithm in dynamic VANET environment.

- Examining the pros and weaknesses of VANET routing protocols.
- Simulation results using NS2 software.
- Various performance metrics, including as energy consumption, drop, throughput, delay, and fairness index, are used to validate simulation results.

2 Literature Review

To maximize the performance of the VANET, Ghaleb et al. [14] introduced the ensemble-based hybrid context-aware misbehavior detection system (EHCA–MDS), which was developed in four phases. By examining the spatial and temporal aspects of the mobility information gathered from neighboring cars, dynamic thresholds were used instead of static thresholds. Furthermore, the technique was built on the Kalman filter, which was used to capture the movement information of the nearby car. The dynamic context reference was then established using the hampel filter. The hampel-based z-score was used to estimate the vehicle's behavior. The cluster-based VANET-oriented evolving graph (CVoEG) architecture was presented by Zahid Khan et al. [15] to maximize the effectiveness of vehicular communications. The eigen gap heuristic was used to divide the VANET nodes, specifically vehicles, into an optimal number of vehicles. The CEG-RAODV reliable routing protocol was introduced as part of the described framework to determine the source to destination reliable journey. Mirsadeghi et al. [16] introduced a trust-based authentication approach-based clustering vehicular ad hoc network to improve road traffic and urban management by detecting rogue nodes and reducing overhead and delay. The clusters were then created to help stabilize the overall network. The trust degree for each vehicle was calculated by combining the trust between vehicles and the trust between road side units (RSUs) and vehicles. Cluster chiefs were chosen based on this level of confidence. The unmanned aerial vehicles (UAV) framework used for recognizing the malicious vehicle, and routing mechanism was given by Fatemidokht et al. [17]. Furthermore, the VRU routing protocol was introduced, which includes two types of routing processes: (i) transmitting data packets between vehicles using the UAV framework and the VRU protocol; (ii) using the described VRU protocol to route data packets among the UAVs. Temurnikar et al. [18] provided the multi-hop clustering method as well as the clustering head selection process for VANET in order to provide dependable interaction among all vehicle nodes. The cluster head was chosen based on the following criteria: Average node speed is closest to node speed, and the cluster head has a long lifespan. They also calculated the QoS matrices and took into account performance measures such as % stability and path length. Ankita Srivastava et al. [19] explored the localization-based routing protocol in VANET to address the challenges that come with it, such as frequent connection failure, packet loss, and latency. They have also used WAVE + LTE technology to get over the limitations of the localization-based routing technique. Talib et al. [20] used the VANETs clustering technique to separate automobiles traveling in a road setting. As a VANETs clustering method, the given work used the center-based evolving clustering based on grid partitioning (CEC-GP) technique. Furthermore, the entire clustering process was integrated, including assigning, cluster head selection, deleting, and merging. Stability, consistency, and efficiency were used to validate the performance of the provided technique. To increase the efficiency of vehicle-to-vehicle communication in VANETs, Bao et al. [21] presented the clustering V2V routing protocol together with the particle swarm optimization (PSO). The cluster heads were picked first, followed by the vehicle nodes and their traveling directions. The fitness function, iteration constraints, route particles, and velocity coding constraints for the presented routing algorithm were then determined. The stream position performance analysis (SPPA) approach was presented by Kolandaisamy et al. [22] to check the position of the field station utilized in delivering the data in order to launch a Distributed Denial of Service (DDoS) attack. Furthermore, the conflict field, conflict data, and attack signature sample rate are all determined using this method (CCA). Through the graph classification with attribute vectors (GCAV) technique, Alaya et al. [23] have developed the multi-objective evolutionary algorithm (MOEA) to improve parameters such as delivery rate, longevity, and minimize inter-class overload. Furthermore, the scalable technique was used to optimize the GCAV's settings. UVANET was secured using a key management technique based on asymmetric and symmetric encryption (KMSUNET), which took into account vehicle node status, vehicle speed, vehicle direction, and the total number of nearest vehicle nodes.

3 Proposed Methodology

This section focuses on the suggested model for detecting malicious activity in the dynamic environment of the VANET, which is based on a hybrid metaheuristic algorithm. The creation of a hybrid model is based on the firefly optimization and whale optimization algorithms, which are referred to as the hybrid firefly with whale optimization algorithm (HFWO). By taking into account, parameters such as distance, distrust value, velocity, and proportional bandwidth, the proposed model enables a highly effective communication paradigm that primarily supports optimizing communication between clusters in the network and assisting in the retrieval of the cluster head. These factors assist in coping with vehicle mobility limitations depending on QoS parameters. The QoS parameters are extremely important when choosing a cluster head to manage the loads by implementing a good distribution plan over the whole network. This protocol can also secure the connectivity and stability of the vehicle network. In this type, the cluster head selection process is first carried out in order to choose or pick the more reliable vehicle as cluster head. The entire fleet of vehicles is involved in this procedure, which elects the cluster head at the same time. Following that, nearby nodes choose gateways to transfer packets and establish clusters based on the cluster head. Finally, in the event of gateway failure, a quick alternate solution for effective transaction through the suggested architecture must be found.

In the proposed VANET concept, all vehicles are equipped with a global positioning system (GPS) that allows the network's geographical information to be retrieved. The firefly optimization approach and the whale optimization technique are both related with the HFWO-VANET model.

3.1 HFWOA-Based Intelligent Clustering

A complex network structure and a high traffic density result in a more complicated network structure in a VANET setting. As a result, an effective clustering strategy is necessary to reduce network complexity while increasing network stability by reducing the number of clusters. In a complicated environment, using the HFWOA-VANET in a minimum clustering setting allows for a less number of cluster heads to be used to improve communication performance. The cluster head selection process is essential for successful communication between nodes or cluster members by transmitting information between them. As a result, by combining two algorithms, such as firefly optimization and whale optimization algorithm, a hybrid metaheuristic algorithm is created. As a result, the HFWO method is used to deliver a set of effective CH solutions in the VANETs environment. Each search agent in the HFWOA-VANET system signifies a solution set with QoS values such as distance, distrust value, and proportionate bandwidth. The QoS parameters are used to optimize the cluster head selection process by ensuring that vehicle loads are dispersed evenly throughout the network. As a result, the network protocol can ensure the stability and connectivity of the vehicle network. The cluster head is responsible for forming the cluster, data collection within the cluster region, transaction between CH and other CHs, load distribution, network stability, and cluster termination. The CHs have direct interface with the base station in this clustering strategy (BS). In addition to the cluster head selection procedure, the neighborhood (gateway) selection process is critical for retransmitting data and establishing a cluster.

Furthermore, each vehicle has a data area for managing neighborhood details, where all of the vehicle's data is originally saved in a standard manner. The malicious node can then be identified using the HFWOA technique and placed in the harmful data space. The data from the neighborhood aids in the execution of improved communication in clusters, preventing information loss. As previously stated, all vehicles manage the neighborhood list using beacon messages with a QoS value generated by each vehicle. The vehicle id, position, mistrust value, and speed are all factored into the QoS value. In reality, the neighborhood list with vehicle QoS values also depicts the availability of a network area's number of neighbor's vehicles. Furthermore, network stability can be adjusted by evaluating vehicle velocity and distance. These characteristics aid in determining the CH and CM for data transport over long distances and at high speeds. This procedure aids in extending the network's life and reducing network communication failures.

The following formulas are considering for determining the velocity and distance of the vehicle. The formula of velocity ration, average speed, velocity, distance ratio, and residual distance is shown in Eq. (1), Eq. (2), and Eq. (3), respectively.

$$\text{Vel\_ratn} = \text{Vel\_C}_{\text{R}}/\text{Avg\_Spd} \tag{1}$$

$$\text{Res\_Dist} = \text{M\_Dist} - \text{CPos} \tag{2}$$

$$\text{D\_ratn} = \text{Res\_Dist}/\text{M\_Dist} \tag{3}$$

where

CPos	Vehicles' current position;
M_Dist	Distance calculation between source and target;
Vel_C_{R}	Velocity ratio;
Avg_ Spd	Average speed;
Vel_C_{R}	Velocity;
D_ratn	Distance ration;
Res_Dist	The ratio of residual distance toward destination.

The other considerable QoS parameter is distrust value that helps to distinguish the behavior of vehicle while transferring hello messages between vehicles to vehicles. This value is essential for vehicle to join in a new network, where when a vehicle joins in a network, its initial distrust value is 1 i.e. $\text{V}_{\text{dis}} = 1$, and the same value is assigned to all vehicles in a group. The entire vehicles distrust value is also stored in neighborhood table.

where,

V_{dis} distrust value of vehicle.

Furthermore, as discussed above, the QoS value is highly significant to calculate the performance of the vehicle using the neighborhood table for analyzing the vehicle's information. The following provides the formula of analyzing QoS value.

$$\text{QoS}_{\text{Veh}} = (\text{Ban}_{\text{Veh}} \times \text{Nei}_{\text{Veh}} \times \frac{\text{D\_ratn}}{\text{Vel\_CR}}\bigg)/\text{V}_{\text{dis}} \tag{4}$$

where

QoS_{Veh}	Vehicle's Qos value;
Ban_{veh}	Available bandwidth;
Nei_{veh}	Number of neighbors;
V	Vehicle.

The suggested HFWO algorithm aids in the analysis of a vehicle's QoS value, and the vehicle with the highest QoS value is designated as a cluster head (Fig. 1).

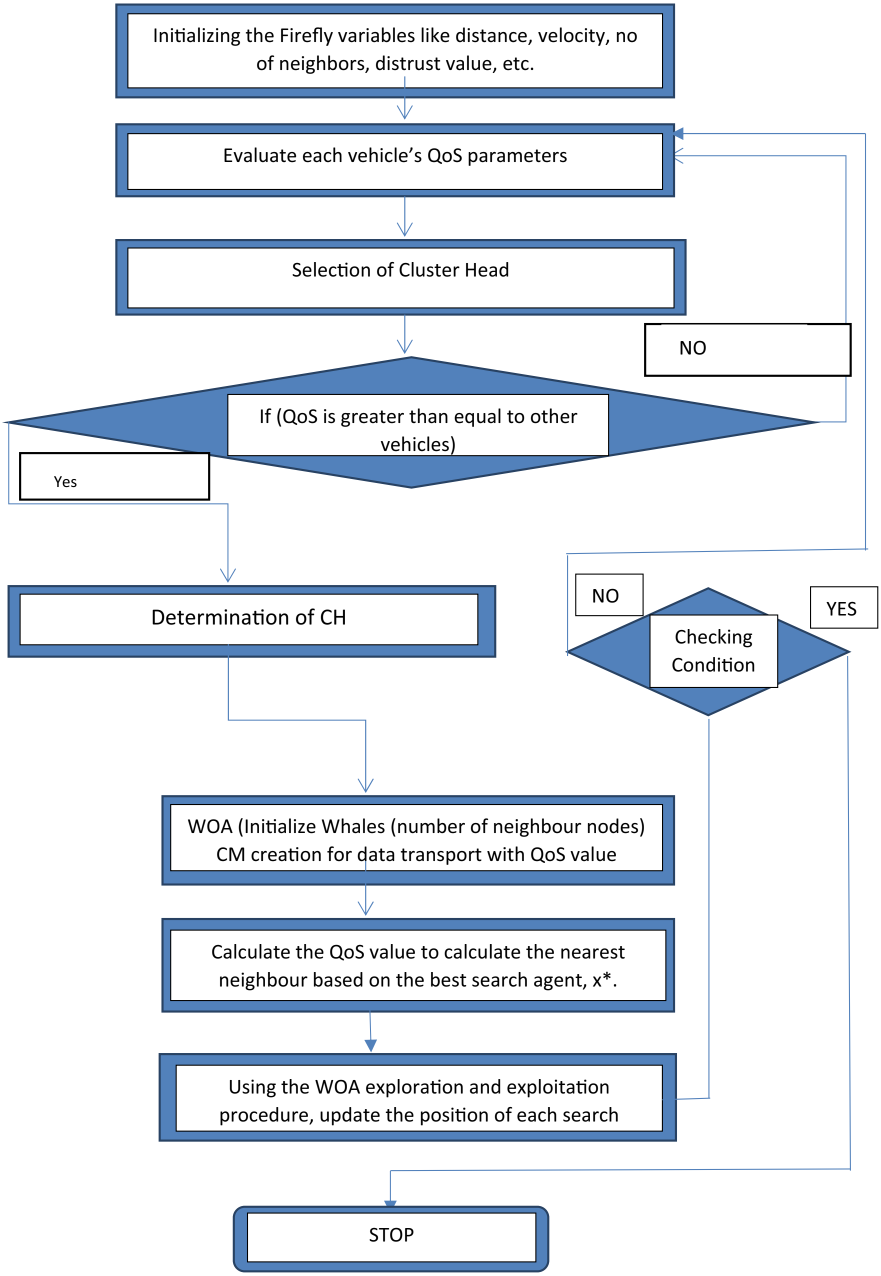

Initializing the Firefly variables like distance, velocity, no of neighbors, distrust value, etc.
Evaluate each vehicle's QoS parameters
Selection of Cluster Head
NO
If (QoS is greater than equal to other vehicles)
Yes
Determination of CH
NO
Checking Condition
YES
WOA (Initialize Whales (number of neighbour nodes) CM creation for data transport with QoS value
Calculate the QoS value to calculate the nearest neighbour based on the best search agent, x*.
Using the WOA exploration and exploitation procedure, update the position of each search
STOP

Fig. 1 Flowchart of proposed model

This process is continued until all of the vehicles in the cluster have been updated, and either a new vehicle has joined or an existing vehicle has left the cluster. The current clusters can be differentiated using HFWOA.

At first, each vehicle is placed inside the white list, and then, if its distrust value becomes larger than a threshold value, it moved to the malicious list. This step ensures the importance of a threshold value for determining and separating malicious node. The following equation shows the condition of the threshold value

$$\delta = e^{\varphi} 0 \leq \varphi \leq K_v - 1 \tag{5}$$

where

K_v average number of vehicles;
N_{avg} average number of vehicles;
T_{avg} typical transmission range.

$$Kv = T_{\text{avg}} / N_{\text{avg}} \tag{6}$$

By analyzing these conditions, the abnormal behavior of the vehicle can be determined. In addition, in this model, the vehicles in a cluster act as verifier which analyze the misbehavior by means of threshold value and distrust values. In case, a vehicle QoS value especially distrust value is determined as larger than the threshold value it is considered as malicious and it directly moved to the malicious vehicle list from normal vehicle list. The following flowchart shows that how the HFWOA algorithm analyzes the QoS value to differentiate the malicious vehicle from normal vehicle.

The next section provides the detailed information about the results obtained through the proposed model. Similarly, a comparative analysis also incorporated in the following section to prove the HFWOA-VANET is highly effective.

4 Results and Discussions

This section discussing the obtained results from the analysis of the proposed technique's performance with the help of utilizing the performance matrices such as energy consumption, drop, throughput, delay, and fairness index (FI). With the utilization of these matrices, the performance of the proposed technique is validated. The experimental process is conducted for proposed and existing techniques and the obtained results are tabulated. The results are attained based on the analysis of performing node-based results. Furthermore, the HFWOA-VANET is the proposed technique, and the QMM-VANET is considered as the existing technique. With the help of the performance matrices, the obtained results are demonstrating that the proposed HFWOA-VANET is performed well when compared to the existing

Table 1 Obtained results of proposed and existing techniques based on node

Proposed HFWOA-VANET					
Node	Energy consumption	Drop	Throughput	Delay	Fairness index (FI)
50	97	0.97895	22,308	0.8105	2
60	97	0.97499	21,097	0.9636	2
70	96	0.96234	23,400	0.9877	1
80	96	0.96080	22,105	1.1370	1
90	96	0.95954	21,100	1.3452	1
Existing QMM-VANET					
50	98	0.99458	18,921	1.2289	1.90784
60	98	0.99365	17,100	1.6831	1.482
70	97	0.98521	16,890	1.8947	1.2962
80	97	0.98620	15,735	2.2365	0.8723
90	97	0.97803	15,265	2.5955	0.6658

QMM-VANET technique. Table 1 illustrating the obtained results of the proposed and existing techniques.

5 Comparative Analysis

This section performing the comparative analysis of the proposed HFWOA-VANET technique with the existing technique named QMM-VANET. The obtained results are tabulated and discussed in the previous section (Sect. 4) which is attained based on node. The graphical representation for the node versuss energy consumption is demonstrated in Fig. 2 in terms of a proposed and existing technique.

As per Fig. 2, the proposed HFWOA-VANET is consumed a low amount of energy when compared to the existing techniques in terms of node. Figure 3 demonstrating the graphical representation of drop vs node for the proposed and existing technique.

The graph which is illustrated in Fig. 3 also shows that the proposed technique has a low drop level compared to the existing techniques in terms of node. Similarly, the graphical representation of throughput versus node is shown in Fig. 4 for the proposed and existing techniques.

In Fig. 4, the graphical representation of throughput versus node clearly shows that the proposed technique has a high throughput value compared to the existing technique. Likewise, Fig. 5 illustrates the graphical representation of delay versus node for the proposed and existing techniques.

As per Fig. 5, the proposed technique has a low delay time when compared to the existing techniques in terms of node. Moreover, Fig. 6 shows the graphical representation of fairness index (FI) versus node. As per Fig. 6, the proposed technique has a high fairness index (FI) when compared to the existing techniques in terms of node.

Fig. 2 Graphical representation for energy consumption versus node

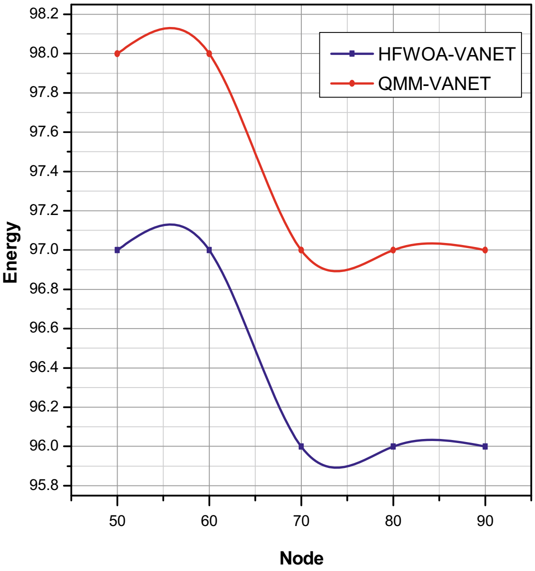
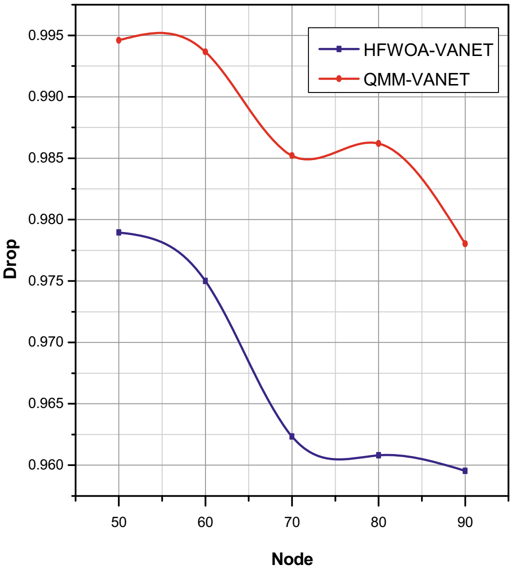


Fig. 3 Graphical representation for drop versus node



The comparative analysis of the proposed HFWOA-VANET with existing QMM-VANET is proved that the proposed technique is a more efficient method for detecting the malicious activity in dynamic VANET environmental information.

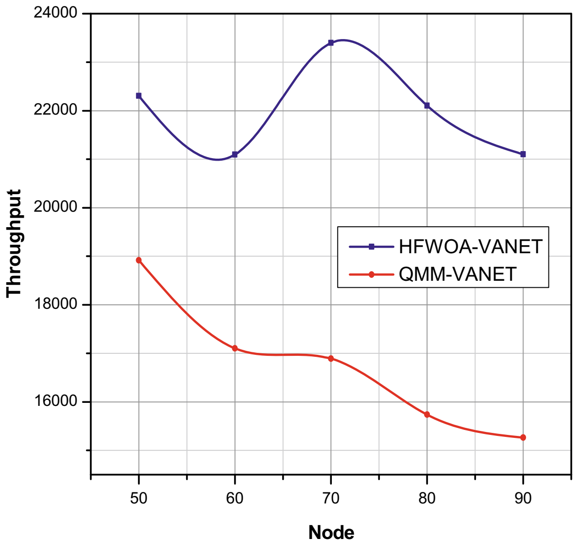


Fig. 4 Graphical representation for throughput versus node

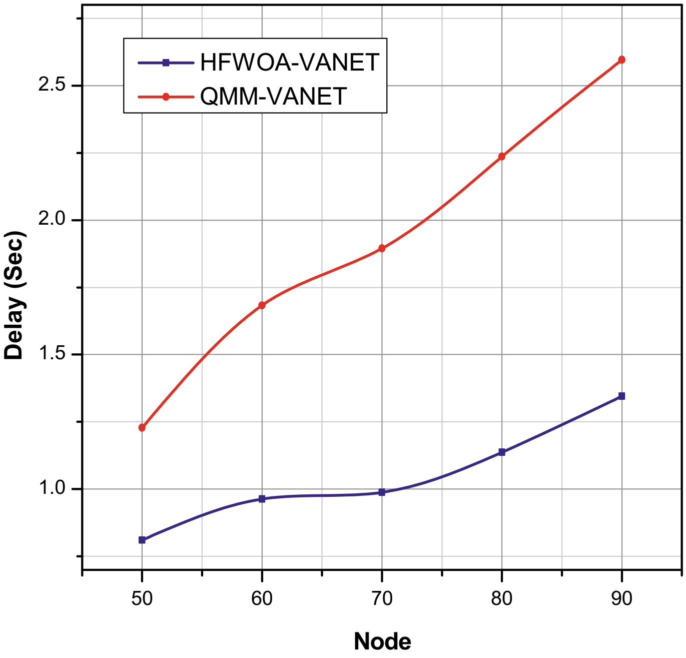


Fig. 5 Graphical representation for delay versus node

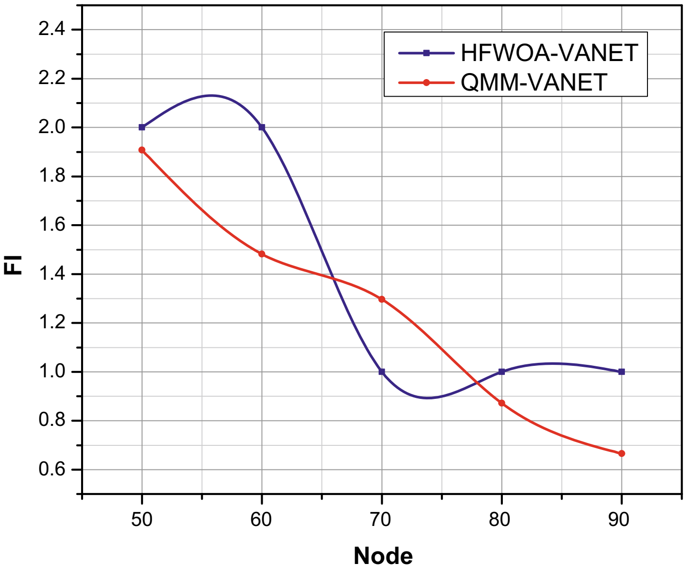


Fig. 6 Graphical representation for fairness index (FI) versus node

6 Conclusion

This work uses VANET technology to prevent malicious vehicle behavior in a VANET environment. The use of hybrid technology to boost the performance of VANET technology and ensure self-organization. In the meantime, it should consider the vehicles' safety by analyzing aberrant vehicle behavior. As a result, the suggested model is built using a hybrid metaheuristic technique that combines firefly optimization with the whale optimization algorithm. A HFWOA-VANET is the result of combining these algorithms. By monitoring data from nearby vehicles, the HFWOA helps to prevent accidents and misbehavior. When the suggested model is compared to the existing QMM-VANET technique, it is clear that the proposed model outperforms the existing technique. The implementation is carried out on the NS2 platform, with performance metrics such as the fairness index (FI), energy usage, drop, delay, and throughput examined. Furthermore, these metrics take into account node variants. As seen from the results, it has been analyzed that average energy consumption has been improved from 97.4 to 96.4, drop from 0.987534 to 0.967324, throughput from 16,782.2 to 22,002, delay has been improvised from 1.92774 to 1.0488, and value of fairness index from 1.244828 to 1.4, respectively.

References

1. Haghighi MS, Aziminejad Z (2020) Highly anonymous mobility-tolerant location-based onion routing for VANETs. IEEE Internet Things J 7:2582–2590. https://doi.org/10.1109/JIOT.2019.2948315
2. Gao H, Liu C, Li Y, Yang X (2021) V2VR: reliable hybrid-network-oriented V2V data transmission and routing considering RSUs and connectivity probability. IEEE Trans Intell Transp Syst 22:3533–3546. https://doi.org/10.1109/TITS.2020.2983835
3. Liang J, Lin Q, Chen J, Zhu Y (2020) A filter model based on hidden generalized mixture transition distribution model for intrusion detection system in vehicle ad hoc networks. IEEE Trans Intell Transp Syst 21:2707–2722. https://doi.org/10.1109/TITS.2019.2905415
4. Yao Y, Xiao B, Yang G, Hu Y, Wang L, Zhou X (2019) Power control identification: a novel Sybil attack detection scheme in VANETs using RSSI. IEEE J Sel Areas Commun 37:2588–2602. https://doi.org/10.1109/JSAC.2019.2933888
5. Rath M, Pati B, Pattanayak BK (2019) Mobile agent-based improved traffic control system in VANET. Springer Singapore
6. Karunakar P, Matta J, Singh RP, Kumar OR (2020) Analysis of position based routing Vanet protocols using Ns2 simulator. Int J Innov Technol Explor Eng 9:1105–1109. https://doi.org/10.35940/ijitee.e2717.039520
7. Aravindhan K, Dhas CSG (2019) Destination-aware context-based routing protocol with hybrid soft computing cluster algorithm for VANET. Soft Comput 23:2499–2507. https://doi.org/10.1007/s00500-018-03685-7
8. Deshmukh AR, Dhawale AS, Dorle SS (2020) Analysis of cluster based routing protocol (CBRP) for vehicular adhoc network (VANet) in Real Geographic Scenario. In: IEEE international conference on electronics, computing and communication technologies (CONECCT), pp 1–5
9. Lee M, Atkison T (2021) VANET applications: past, present, and future. Veh Commun 28:100310. https://doi.org/10.1016/j.vehcom.2020.100310
10. Al-Shareeda MA, Anbar M, Manickam S, Yassin AA (2020) VPPCS: VANET-based privacy-preserving communication scheme. IEEE Access 8:150914–150928. https://doi.org/10.1109/ACCESS.2020.3017018
11. Mamatha T, Aishwarya P (2019) An efficient cluster based routing protocol using hybrid FCM-Q LEACH for vehicular ad hoc networks. Int J Appl Eng Res 14:1604–1612
12. Shrivastava PK, Vishwamitra LK (2021) Comparative analysis of proactive and reactive routing protocols in VANET environment. Meas Sensors 16:100051. https://doi.org/10.1016/j.measen.2021.100051
13. Oche M, Tambuwal AB, Chemebe C, Noor RM, Distefano S (2020) VANETs QoS-based routing protocols based on multi-constrained ability to support ITS infotainment services. Springer, US
14. Ghaleb FA, Maarof MA, Zainal A, Saleh Al-Rimy BA, Alsaeedi A, Boulila W (2019) Ensemble-based hybrid context-aware misbehavior detection model for vehicular ad hoc network. Remote Sens 11. https://doi.org/10.3390/rs11232852
15. Khan Z, Fan P, Fang S, Abbas F (2019) An unsupervised cluster-based VANET-oriented evolving graph (CVoEG) model and associated reliable routing scheme. IEEE Trans Intell Transp Syst 20:3844–3859. https://doi.org/10.1109/TITS.2019.2904953
16. Mirsadeghi F, Rafsanjani MK, Gupta BB (2020) A trust infrastructure based authentication method for clustered vehicular ad hoc networks. Peer-to-Peer Netw Appl. https://doi.org/10.1007/s12083-020-01010-4
17. Fatemidokht H, Rafsanjani MK, Gupta BB, Hsu CH (2021) Efficient and secure routing protocol based on artificial intelligence algorithms with UAV-assisted for vehicular ad hoc networks in intelligent transportation systems. IEEE Trans Intell Transp Syst 22:4757–4769. https://doi.org/10.1109/TITS.2020.3041746
18. Deepthi P, Sivakumar S, Murugesan R (2017) Development of Multihop clustering algorithm for the simulation of VANET. Int J Comput Appl Math 12:562–568

19. Srivastava A, Prakash A, Tripathi R (2020) Location based routing protocols in VANET: Issues and existing solutions. Veh Commun 23:100231. https://doi.org/10.1016/j.vehcom.2020.100231
20. Talib MS, Hassan A, Alamery T, Abas ZA, Mohammed AAJ, Ibrahim AJ, Abdullah NI (2020) A center-based stable evolving clustering algorithm with grid partitioning and extended mobility features for VANETs. IEEE Access 8:169908–169921. https://doi.org/10.1109/ACCESS.2020.3020510
21. Bao X, Li H, Zhao G, Chang L, Zhou J, Li Y (2020) Efficient clustering V2V routing based on PSO in VANETs. Meas J Int Meas Confed 152:107306. https://doi.org/10.1016/j.measurement.2019.107306
22. Kolandaisamy R, Noor RM, Kolandaisamy I, Ahmedy I, Kiah MLM, Tamil MEM, Nandy T (2021) A stream position performance analysis model based on DDoS attack detection for cluster-based routing in VANET. J Ambient Intell Humaniz Comput 12:6599–6612. https://doi.org/10.1007/s12652-020-02279-2
23. Alaya B, Sellami L (2021) Clustering method and symmetric/asymmetric cryptography scheme adapted to securing urban VANET networks. J Inf Secur Appl 58:102779. https://doi.org/10.1016/j.jisa.2021.102779

Wireless Networks and Internet of Things (IoT)

Localization of Sensor Node by Novel Quantum Walk-Pathfinding Rider Optimization (QWPRO) by Mobile Anchor Node

Om Mehta and Seema Mahajan

Abstract Wireless sensor network (WSN) is used in many fields but the problem in that is the exact location of the sensor nodes is not known to the main user. For this the only solution is to use the GPS on sensor node; but the option makes the system an energy inefficient one and also, high-cost option. Hence a beacon node is introduced in the network with the GPS so that the location of the beacon node is controlled by the user. The beacon node estimates the position of the sensor node on the network field by using several concepts; but the localization accuracy is failed to satisfy the requirement. Hence in this research work, the mobile beacon node is used in the network to locate the sensor node. The trajectory of the beacon node is set as zigzag and the beacon node takes the quantum walk on the network by employing the glued tree concept to locate the node. The proposed model is the range-based localization technique utilizing the received signal strength indicator (RSSI) and time of arrival (ToA) ranging of the nodes. Then the coordinate of the node in the field is set by the novel pathfinding rider optimization (PRO). After the analysis, the average localization error of 0.002% is obtained for the 100 sensor nodes.

Keywords Wireless sensor network · Mobile node · Localization · Received signal strength indicator · Time of arrival · Beacon node · Random walk · Quantum walk · Pathfinder optimization · Rider optimization

1 Introduction

Wireless sensor networks (WSN) can be used in different applications such as health monitoring, underwater acoustic monitoring, industrial monitoring, environment monitoring, etc. on all these applications, sensor networking is different [1–3]. The easy deployment and organization of the network make the system more suitable and adaptable to many applications. For varied applications, the network demand is

O. Mehta · S. Mahajan (✉)
Indus University, Ahmedabad, India
e-mail: ce.hod@indusuni.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_10

different, which is based on hierarchical structure, heterogeneous network, homogeneous network, static or mobility of sensor node [4–8]. The nodes will gather the information on various environments and deliver the information to the destination [9]. To evaluate the environmental properties, a varied types of sensor nodes are used such as optical, biomedical, chemical sensors, etc. Some environment requires the heterogeneous network, or homogenous network, some requires static or mobile nodes they are based on the requirement of the application model. But, one thing is same for every application which is the localization of the sensor node [10]. The sensor node is operated by the battery and is made by the cost-efficient one, so the placement of GPS system on the sensor node will make the whole network pricey [11] along with the faster reduction in energy level of the sensor node. Hence the location of the node is difficult to monitor its movement either in static as well as in mobility. The estimation of the position of sensor node on either case is difficult to forecast.

In order to know the geographic location of the sensor node, the GPS system can be equipped with the sensor or place a static sensor manually on the field [12, 13]. But the sensor equipped with the GPS receiver will form a high-cost system model and the manually placing the sensor is not applicable for every application and is inefficient. Hence the researchers focused on the localization problem of sensor node and divide the methodology as centralized techniques and distributed techniques. On the centralized technique, the base station itself evaluates the position of the nodes in terms of distance [14]. Then the calculated distance is transferred to the appropriate node. The technique provides better accuracy rate for the localization but the drawback is the lack of capacity to hold the data, which tends to incomplete information transmission [15]. Due to this drawback, the method is not used for the large-scale sensor network that is the method is applicable only for the specified applications. Besides, the distributed technique or self-localization [16] evaluates the required measurement by employing the anchor or beacon nodes in the network. The measurement indicates the noise, distance, or any other parameter to evaluate the location.

The Bayesian-based approaches [17] provide solution to the localization problem by measuring the noise on the received signal; inhibiting the uncertainty prediction of the location named as belief [18]. The problem of the filtering method is it requires separate particle filters to have clear information about the belief [17, 19]. To overcome the issues related to the filtering model, the technique is further classified as range-based and range-free technique [20]. The range-based localization is based on the angle of the adjacent nodes and the distance between the nodes to find the location of the specific node. The range-based method uses the time difference of arrival (TDoA), time of arrival (ToA), received signal strength indicator (RSSI), and angle of arrival (AoA) to find the distance between the nodes. By using any of these methods the location of the sensor node is detected. The range-free localization model does not use distance or angle information among the nodes but makes use of proximity information [21]. The approach works by network coverage and the contents of received messages. The model uses centroid method, distance vector hop (DV-Hop), amorphous algorithm, etc. [22] to provide the position of the sensor node. When compared to the range-based method, the range-based method provides error-free localization, whereas, the range-free method consumes low power on the network.

2 Literature Survey

There are several research done by many researchers to solve the localization problem of WSN by using both range-based and range-free techniques. In general most of the techniques uses the RSSI to locate the nodes so that the network and the signal delivery rate can be enhanced. Range-based and range-free techniques used so far are illustrated in this section.

By means of range-based technique several models were designed by considering the trajectory of anchor node, number of anchor node, RSSI, and also as a case study which are as follows: To overcome the problem of optimum path selection for the mobile anchor node to localize the sensor node, Kannadasan et al. [12] proposed the M-curve model. The localization of the sensor node was performed by the Dolphin Swarm Algorithm (DSA). In that model, every node receives three non-collinear beacon signals, which overcome the problem of collinearity. The trajectory fulfills the coverage and localization issues. Further reduction in the localization error was made by the DSA. Another model that focuses on path to the mobile anchor node was developed by Thilagavathi and Manickam [23] as enhanced RSSI-based tree climbing (ERTC) model. Here first the distance of the node was calculated under the phenomenon that distance was inversely proportional to the RSSI. Based on that the anchor node makes a tree climbing mechanism to locate the nodes position. For the localization of static sensor nodes, three mobile anchor nodes were used by Ibrahim et al. [24] which replaces the use of GPS unit on sensor. The developed model was called as triple mobile anchors for localization (TMAL). Those anchor nodes were placed adjacent to each other forming a triangle having the sensing range of its radius. The mobile anchor node would use the RSSI to locate the node and was a rechargeable system. Due to the use of rechargeable units in three mobile nodes, the difficult arises in outdoor applications.

Majority of the researches have considered the localization problems in 2D plane, but Singh and Mittal [25] proposed the localization scheme for the 3D plane by infusing the adaptive flower pollination algorithm (adaptive FPA). The model works by employing single anchor node and multiple virtual anchor nodes on the network. The anchor node position the unknown node by targeting the virtual node. After that to meet the localization in 3D sparse network, Liu et al. [26] proposed a model for the sparse 3D sensor network. In 2D the sparse network issue was resolved by Zhang et al. [27] using patch and sitting localization strategy. But in 3D model, the network conditions were derived to merge two sub-network. To merge those networks both common nodes and adjacent edges were considered.

2.1 Problem Statement

The sensor nodes have the ability to sense the information from where it is placed and to transmit that gathered information to the user or human operator. The sensing capability of the sensor is used in WSN, which can be applied for many applications. But based on the application the sensor nodes may be static or dynamic along with that the detailed information gathered by the sensor is necessary. The detailed information indicates the position from which the data are acquired by the sensor. Also before initiating the data transmission, the location at which the information is gathered needs to be known to the nodes, which should affix to the data.

But the sensor nodes are always not used with the location tracking systems like GPS since that will cost a lot. And also it is not an easy manufacturing process of the sensor node since the use of GPS on nodes will induce heavy energy consumption rate. The WSN is most widely used in hard environmental conditions, so the position of the nodes will get changed. Along with that the process of avoiding the sensor hub will cause treat in the communication process by means of attack. The level of localization have a poof for the security issues on the network. Hence the localization should bring the security alert for the network.

3 Proposed Methodology

The methodology of the paper is to present a model to locate the sensor on the network field that can be applied for any application. The objective of the glued tree model in the work is to have a trajectory for the movement of the mobile beacon node on the network. The spiral trajectory of the beacon node have the impact due to the missing localization of the nodes at the corner on the network field also, the straight line traversing yield the collinear points. Hence in this paper, the zigzag traversing of the mobile beacon node is imposed. The traversing is done by considering the RSSI, ToA as well as the distance of the node from the beacon node. When the distance is low, the signal strength is maximum, which is an essential for the signal transmission. Also, the ToA will work better during the presence on LOS; hence by considering these two factors the mobile node will localize the sensor node on its sensing radius. To achieve this, the tree structure is the best option for the movement of beacon node. The beacon node localizes the sensor nodes position when the RSSI is high; this will again reduce the error of localization. The proposed structure of node localization is as follows.

The mobile beacon node traverses on the network area in the zigzag trajectory by means of the glued tree concept employing the continuous quantum walk. The node having high RSSI will get the high priority of localization.

When the RSSI will get lowered for the node at the edge of beacon node radius, then the ToA of the node is calculated for the node with LOS on the beacon node sensing radius.

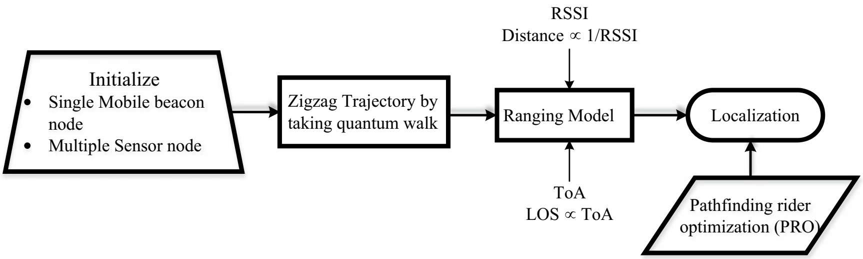

RSSI
Distance ∝ 1/RSSI
Initialize
Single Mobile beacon node
Multiple Sensor node
Zigzag Trajectory by taking quantum walk
Ranging Model
Localization
ToA
LOS ∝ ToA
Pathfinding rider optimization (PRO)

Fig. 1 Overall Work of the proposed model

The node having the shortest distance that is low ToA with respect to the beacon node will get localized first. The selection of the shortest distance concerning the beacon node not only depends on ToA, also on RSSI.

Figure 1 shows the overall workflow of the proposed model. Then the PRO is used to locate the sensor node, which has highest RSSI and low ToA. The coordinates of the sensor node are projected by this algorithm. Thus the entire designed model is referred as QWPRO.

3.1 System Model

The system model consists of number of sensor node and a single mobile beacon node. There are many studies that have used number of beacon nodes to locate the unknown nodes but increasing the beacon node will increase the cost of the network since the node is inbuilt with the GPS system. The location of the sensor nodes are unknown and is scattered on the network field, whereas the location of the mobile beacon node is known to the user and is controlled by the human operator. The sensor nodes are deployed on the field and they will stay at their position to do the sensing task. The beacon node is considered to have a GPS system to track the position of the corresponding node thereby controlling it to know the location of other nodes. The beacon node has the sensing radius of R on the network which is useful for broadcast. Since the mobile beacon node is used, the localization can be performed for both static and mobile sensor nodes.

The movement of the mobile beacon node on the network field is based on the wireless communication with the radio propagation model. The sensor within the distance range of the beacon node will have will receive the message transmitted by it and inform its location. The beacon node is modeled with the sufficient energy to do the transmission process by inhibiting the localization. The mobile beacon node is controllable in terms of speed so that they can move in the field as per the controllable speed by the user. When the beacon node broadcast the beacon signal, the receiving nodes are subjected to calculate the distance and LOS from the node to initiate the localization. Basically, the signal strength depends on the distance between both the transmitted and received node whereas, the ToA depends on the LOS condition.

3.2 Mobility Model of Beacon Node

The mobile beacon node will move on the network in the zigzag fashion, which is controlled by the glued tree graph. The glue tree graph that works by taking the continuous quantum walk instead of the random walk is considered for the beacon node movement.

To define the quantum walk, first the random walk model is considered by a graph G with vertices V and edges E. the adjacent matrix A of graph is

$$A_{a,b} = \begin{cases} 1 \text{ if}(a, b) \in E, \\ 0 \quad\quad \text{else.} \end{cases} \tag{1}$$

The Laplacian for random walk is,

$$L_{i,j} = \begin{cases} -\text{degree}(a) & 0 \\ 1 & \text{if}(a, b) \in E, \\ 0 & \text{else.} \end{cases} \tag{2}$$

This Laplacian equation determines for the random walk model having probability, p(t) with length |V|. The differential equation for the probability is given by,

$$\frac{\mathrm{d}}{\mathrm{d}t} p_a(t) = \sum_{(a,b)\in E} L_{a,b} p_b(t). \tag{3}$$

The Schrödinger equation is considered as $a\frac{\mathrm{d}}{\mathrm{d}t}|\psi\rangle = H|\psi\rangle$. Here, H is the Hamiltonian operator. To preserve the normalization state the Laplacian is used. Then the differential equation is given by,

$$a\frac{\mathrm{d}}{\mathrm{d}t}\psi_a(t) = \sum_{(a,b)\in E} L_{a,b}\psi_b(t). \tag{4}$$

where, $|\psi(t)\rangle = e^{\text{-iLt}}|\psi(0)\rangle$. This equation evaluates the analog of continuous quantum walk. The random walk makes the agent to get collide to overcome that quantum walk is used. It is not necessary to define the quantum walk by Laplacian but the structure graph itself defines it, which is done by glued tree graph here.

The quantum walk is an analog of classical random walk, which is caused due to the superposition of the walker (beacon node in the work). The quantum walk has a huge advantage in the quantum algorithm or computation and quantum simulation environments. The quantum walk in one dimension has a worthier property, by using those qualities the quantum walk is used in the presented paper for the 2-dimensional movement of the node in the network. The quantum walk is used in multi-disciplinary researchers but here the quantum walk is used in the tree concept to make an appropriate movement of the node. The quantum states are given by the Hamiltonian operator. The Hamiltonian operator is given by,

$$H\omega = E\omega \tag{5}$$

where, ω gives the velocity and the location and E is the energy level of the beacon node since the movement of beacon node depends on the energy available in it. The mobile beacon node is modeled in such a way that it possesses enough energy to move over the space for locating the unknown nodes. The velocity of the node depends on the energy available, hence by means of the available energy.

The glued tree has set of agents, which takes the continuous quantum walk to reach the destination. The glued tree has one entrance and one exit path within these two paths, the agent will take quantum walk to reach the destination. On glued tree with continuous quantum walk method, the agent in each division is indicated by column. The number of states in each column is C_b with 2b where, b ranges from [0, n] and the number of states is 2^{2n+1-b} then b ranges from [n + 1, 2n + 1]. All the states are placed at equidistance from the entrance and exit nodes. Here n is the height of the tree graph and very state in the graph has two edges. The root agent takes the superposition state at each column, thus making number of states as,

$$|\mathrm{col}\, b\rangle = \frac{1}{\sqrt{C_b}} \sum_{e \in \mathrm{col}\, b} |e\rangle. \tag{6}$$

For the quantum walk, the normalized state is given by, $\frac{1}{\sqrt{C_j}}$. The adjacent matrix A is given by the Hamiltonian for glued tree graph and is given by,

$$A|\mathrm{col}\, b\rangle = \sqrt{2}|\mathrm{col}\, b - 1\rangle + \sqrt{2}|\mathrm{col}\, b - 1\rangle. \tag{7}$$

When $b = n + 1$ is considered to meet symmetricity, the graph is similar to the quantum walk with finite line. $\sqrt{2}$ is the weight at the edges apart from column n and $n + 1$. For n and $n + 1$ the weight is 2. In case of infinite line, the probability of zigzag movement of node is performed at a distance of $2n + 1$ with state $\frac{b}{\mathrm{poly}\, n}$.

At every movement interval, the mobile beacon node broadcast a QUERY message to the nodes which are at the range of anchor node. The nodes at its sensing radius send a message to the beacon node to localize. At this point, all the node on the range are not get localized that depends on RSSI and ToA of the node corresponding to the beacon node.

3.3 Ranging Model

The ranging of the mobile beacon node and the sensor node is based on the RSSI and ToA. Under the communication radius R of the beacon node, by setting the distance r, the RSSI of the nodes are calculated. This step will make access to high RSSI node. Beyond this threshold distance, the LOS nodes under the communication radius is selected and derive the ToA of the node for the localization. This step is clearly illustrated in Fig. 2.

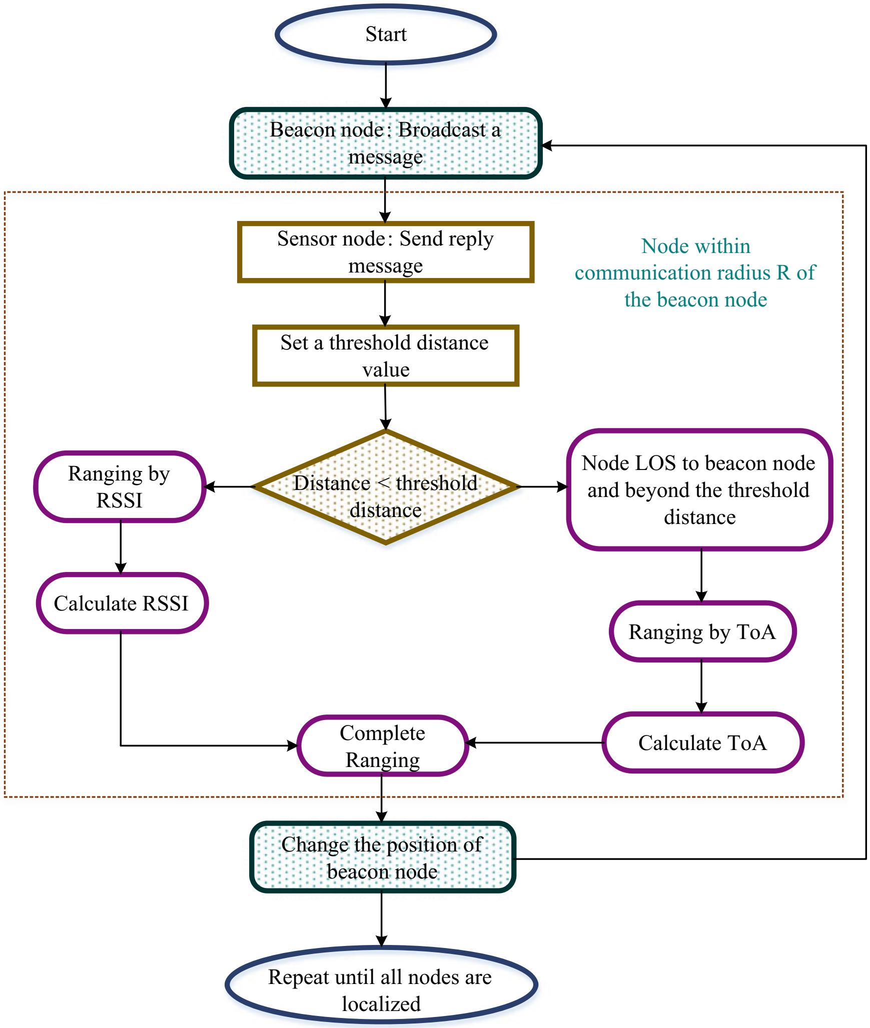

Start
Beacon node: Broadcast a message
Sensor node: Send reply message
Node within communication radius R of the beacon node
Set a threshold distance value
Ranging by RSSI
Distance < threshold distance
Node LOS to beacon node and beyond the threshold distance
Calculate RSSI
Ranging by ToA
Complete Ranging
Calculate ToA
Change the position of beacon node
Repeat until all nodes are localized

Fig. 2 Ranging model for mobile beacon node based on RSSI and ToA

3.3.1 RSSI Calculation

The RSSI of the node is calculated by considering the distance among the node and beacon node. The relation between the distance and RSSI is when distance increases the RSSI of the node get decreased and vice versa. When a node transmits its message the beacon node calculates its signal strength, which depends on distance. The received signal power at the distance r is given by,

$$\mathrm{RS}_p(r) = \mathrm{RS}_p(r_0) - 10\varsigma \log_{10}\frac{r}{r_0} + N \tag{8}$$

where, RS_p is the received signal power at distance r, r_0 is the reference distance of the node. The path loss for the node at the transmitting channel is ς and N is the noise that gets added in the received signal having mean zero and standard deviation σ. All these parameters are defined between the sensor node and the beacon node.

The density function of the signal having the noise other than Gaussian is given by,

$$f(z) = \frac{1}{\sqrt{2\pi\sigma}} e^{(z-\mu)^2/-2\sigma^2} \tag{9}$$

$$\mu = \frac{1}{m}\sum_{s=1}^{m} SI_s \tag{10}$$

$$\sigma = \sqrt{\frac{1}{m-1}\sum_{s=1}^{m}(SI_s - \mu)^2} \tag{11}$$

where, $\mu\ and\ \sigma$ are the mean and standard deviation of the received signal. SI_s is the signal strength of the sth signal and m is the total number of measurements. When the distance ranging increases the RSSI also increases hence the RSSI is measured only at the specified distance D.

3.3.2 Time of Arrival

The beacon node calculates the LOS nodes at one point within the communication radius R of beacon node. If the RSSI of the unknown node fails the ToA of the node is taken for the localization. Then the coordinates of the nodes are set by the PRO. The ToA is calculated by the general equation,

$$\tau = \upsilon.d \tag{12}$$

where, υ is the velocity of the signal transmitted from the sensor node, which needs to get localized. d is the distance and τ is the data propagation time. Consider a

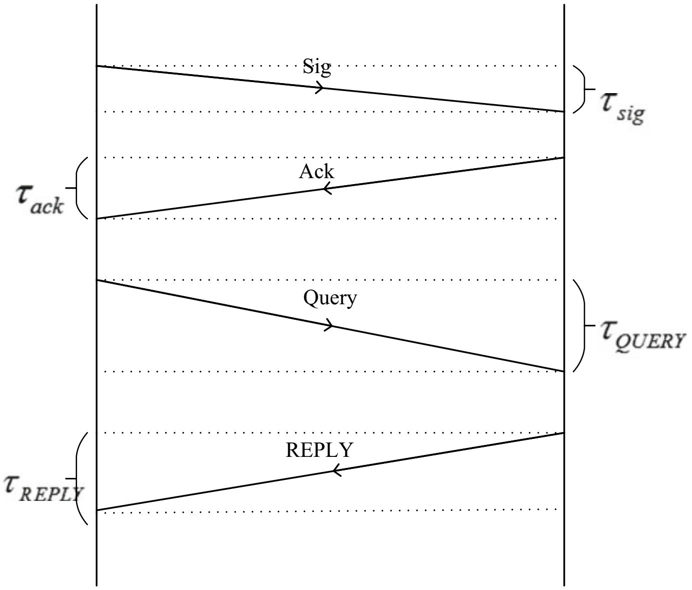


Fig. 3 ToA for the communication among the beacon node and sensor node

transmission scenario of QUERY message transmitted by the beacon node and the REPLY send by the sensor node. First the beacon node makes a beacon signal, then the senor node acknowledges the signal, after that the beacon node sends the QUERY message the sensor node responds by sending the REPLY message. This scenario occurs only to the node that are in LOS with the mobile beacon node. Then ToA at this condition is derived as,

$$\tau_{\mathrm{ToA}} = \frac{\tau_{\mathrm{sig}} + \tau_{\mathrm{ack}} + \tau_{\mathrm{QUERY}} + \tau_{\mathrm{REPLY}}}{4} \tag{13}$$

Here, τ_{sig} is the signaling time for the beacon node, τ_{ack} is the acknowledgment time, τ_{QUERY} is the QUERY message transmission time, and τ_{REPLY} is the REPLY message transmission time and the communication flow is shown in Fig. 3. The Eq. (13) gives the average time taken for the signal to get arrived at the receiver.

3.4 Localization of Sensor Nodes

In this paper, the sensor nodes are localized by the hybrid optimization algorithm. The rider optimization is used to track the location of the sensor node and the indexing of the parameter in the optimization is controlled by the pathfinder optimization. The rider optimization works by considering four groups of riders who targeted to reach the goal. Those groups include bypass rider, overtaker, follower, and attacker. Anyhow, all these groups of rider depends on the leading riders' position, hence the leading rider position is updated by the pathfinder optimization. This method is generally named as PRO.

The number of groups in the optimization is four, which is represented as G having random position for the group. The four groups in the optimization is considered as anchor node (bypass rider), node that is in LOS (follower), node having highest RSSI (overtaker), node having highest RSSI but curtained by other nodes (attacker). The position or the coordinates of all these node settings have to be evaluated, which can be made by the rider optimization. The parameters like steering angle, gear, accelerator, and brake indicate the nodes orientation, displacement, obstacles, and fading of the signal corresponding to the node, respectively. The group or node initialization is given by,

$$x^l = \left\{x_i^l(p,q)\right\}. \tag{14}$$

where, the value for i ranges as $1 \le i \le N$, which is the number of riders in the field and the dimension range is $1 \le p, q \le M$ having p, q as dimension. $x_i^l(p,q)$ is the position of ith rider at time l. The total number of riders is given by summing number of bypass riders, overtakers, attackers, and followers. All those riders in different groups are of one-by-fourth of the total riders. The position of each group lies between the range as $\left[x_1, x_{N/4}\right]$, $\left[x_{N/4+1}, x_{N/2}\right]$, $\left[x_{N/2+1}, x_{3N/4}\right]$, and $\left[x_{3N/4+1}, x_N\right]$ for bypass rider, follower, overtaker, and attacker, respectively. The other initialization parameters include steering angle, gear, accelerator, and brake are also initialized as follows:

The steering angle is given by,

$$\mathrm{SA}_i(p,q) = \begin{cases} \varphi_i & \text{if } p, q = 1, \\ \mathrm{SA}_i(p-1,q-1) + \theta & \text{if } p, q \ne 1 \& SA_i(p-1,q-1) \\ & +\theta \le 360, \\ \mathrm{SA}_i(p-1,q-1) + \theta - 360 & \text{else.} \end{cases} \tag{15}$$

where, φ_i is the position of the ith rider vehicle, which depends on the number of riders and maximum angle and is given by,

$$\varphi_i = i * \frac{360^\circ}{N} \tag{16}$$

The coordinate angle is given by θ and can be determined by the following equation,

$$\theta = \frac{360^\circ}{M} \tag{17}$$

Then the gear, accelerator, and brake of the ith rider is represented as GE_i, AC_i, and BK_i. The value ranges from the set {0, 1, 2, 3, 4}, , (0, 1), and [0, 1], respectively. At initial stage, the gear and accelerator value is 0 whereas the brake is 1. Then the speed is given by,

$$V_i^{\mathrm{GE}} = \frac{(x_i^{\max} - x_i^{\min})/\mathrm{OT}}{|\mathrm{GE}|} \tag{18}$$

where, $x_i^{\max}$, $x_i^{\min}$ is the maximum and minimum value in ith rider position, OT is the off time and is referred as maximum iterations. |GE| is the number of gear.

The success rate that is the destination reaching rate is given by the equation,

$$\mathrm{sr}_i = \frac{1}{x_i - x^I} \tag{19}$$

Here, x^I is the position of the leading rider (position of the current localizing node). The position of the target is already found by the RSSI and ToA conditions. Only the coordinate setting is acquired in this phase. The position of the leading rider is given by the pathfinder optimization. The pathfinder optimization works by considering the swarming behavior of the animal or common action of an individual. The leader is the one who took the swarming by looking for the prey or finding the feeding area. The food searching or searching the place to protect themselves from an issue is considered in the optimization. Here the pathfinder is considered to be the leading rider and its position is given as,

$$x_r(l + \Delta l) = x_r(l) + \Delta x + A \tag{20}$$

where, x_r is the position vector of leading rider, Δx is the distance acquired by the leading rider at l time interval and A is the disturbance that may cause during the riding, which includes wind, obstacles, etc. For every iteration, there is a chance for the leading rider may get changed from leading position. Hence the above equation is modified as,

$$x_i^{k+1} = x_i^k + P \cdot \left((x_p^k - x_i^k) + (x_q^k - x_i^k)\right) + \delta = I \tag{21}$$

Here, k represents the number of iteration, x_p and x_q is the position vector coordinates, δ is the disturbance rate, and P is the random vectors that is used to avoid the collision between the neighbor rider.

Then the position update by the group of rider is given as follows. The bypass rider position on both coordinates is represented as,

$$\begin{aligned} x_{B,i}^{l+1}(p) &= \alpha\left[x^l(\eta, p) * \rho(p) + x^l(\xi, p) * [1 - \rho(p)]\right] \\ x_{B,i}^{l+1}(q) &= \alpha\left[x^l(\eta, q) * \rho(q) + x^l(\xi, q) * [1 - \rho(q)]\right] \end{aligned} \tag{22}$$

where, η and ξ depends on N in the range of 1 to N and α ρ lies between 0 to 1. The position update performed by the follower is given by,

$$\begin{aligned} x_{F,i}^{l+1}(p') &= x^{I}(I, p') + \left[\cos(SA_i(p')) * x^{I}(I, p') * d_i^l\right] \\ x_{F,i}^{l+1}(q') &= x^{I}(I, q') + \left[\cos(SA_i(q')) * x^{I}(I, q') * d_i^l\right] \end{aligned} \tag{23}$$

where, (p', q') are the coordinates of the follower and I is the indexing of the leading rider whereas, x^I is the position of the leading rider. The distance d_i^l is measured by the velocity of the rider with off time as per the Eq. (18). The position update by the overtaker is given by,

$$\begin{aligned} x_{O,i}^{l+1}(p') &= x_i^l(p') + \left[\mathrm{DI}_i^l * x^{I}(I, p')\right] \\ x_{O,i}^{l+1}(q') &= x_i^l(q') + \left[\mathrm{DI}_i^l * x^{I}(I, q')\right] \end{aligned} \tag{24}$$

Here, x_i^l is the position of ith rider at coordinates (p', q') DI_i^l is the direction indicator for the ith rider at time l. The direction indicator is given by,

$$DI_i^l = \left[\frac{2}{1 - \log\left(SR_i^{r,l}\right)}\right] - 1 \tag{25}$$

$SR_i^{r,l}$ is the relative success rate, which is obtained by the ratio of success rate of a rider to the maximum success rate attained by N rider. The direction indicator will have the value ranges between -1 to 1. The position update by the attacker is given by the Eq. (26),

$$\begin{aligned} x_{A,i}^{l+1}(p) &= x^{I}(I, p) + \left[\cos(SA_i(p)) * x^{I}(I, p) * d_i^l\right] \\ x_{A,i}^{l+1}(q) &= x^{I}(I, q) + \left[\cos(SA_i(q)) * x^{I}(I, q) * d_i^l\right] \end{aligned} \tag{26}$$

The above equation is the position of attacker on the coordinates (p, q) from the leading rider position x^I with the distance d_i^l at time interval l. Thus the sensor node coordinates are set by the hybrid optimization algorithm.

```
Algorithm
Step 1: Initialize the number of riders population along with the parameters like $SA$, $GE$, $AC$, and $BK$
Step 2: Calculate the success rate by using the Eq. (19)
Step 3: Maximum iteration is given by off time $OT$//Pathfinder optimization
Step 4: Calculate the position of leading rider by Eq. (21)//Rider optimization
Step 5: For N number of riders do
Step 6: Update the position of bypass rider by Eq. (22)
Step 7: Update the position of follower by Eq. (23)
Step 8: Update the position of overtaker by Eq. (24)
Step 9: Update the position of attacker by Eq. (26)
Step 10: Evaluate the position with the success rate
Step 11: Select the rider having the maximum success rate as leading rider
Step 12: Update the position of leading rider
Step 13: Increment the iteration $OT$
Step 14: End for
Step 15: Terminate
```

4 Result Analysis

The analysis of the proposed localization model is implemented in the MATLAB platform. 100 s of sensor are placed on the network area of 100 × 100 m, which are localized by the single mobile beacon node. The beacon node move on the network field in the zigzag pattern for the localization of sensor node. The communication radius of beacon node is 10, 20, 30, 40, and 50 m. By varying the radios the analysis is done. The average distance to track the RSSI of the node is set as 20 m. The quantum walk taken by the beacon node would allow it to have the optimized selection of nodes to get localized considering the RSSI and ToA. The communication among the mobile node and the sensor node is made by the temporarily ordered routing algorithm (TORA). Initially, the anchor node is deployed on the top-left corner of the field. The RSSI and ToA for the node corresponding to the beacon node is estimated after that, the PRO is employed with the population density of 100 having OT as 500.

The movement model of the mobile beacon node is shown in Fig. 4. Here the triangle shows the mobile beacon node position at every localizing state. At every state, the beacon node calculates the RSSI and ToA based on the proposed model condition for the approval of the node to get localized. In figure, the red colored line indicates the trajectory path of the mobile beacon node.

Figure 5 illustrates the nodes that are at the communication radius R of the mobile beacon node. At every position, it takes the nodes that are associated with the RSSI and ToA conditions provided to the model.

The coordinates of the sensor node based on the RSSI and ToA-based ranging are set by the PRO algorithm. The localization error perceived by using that algorithm

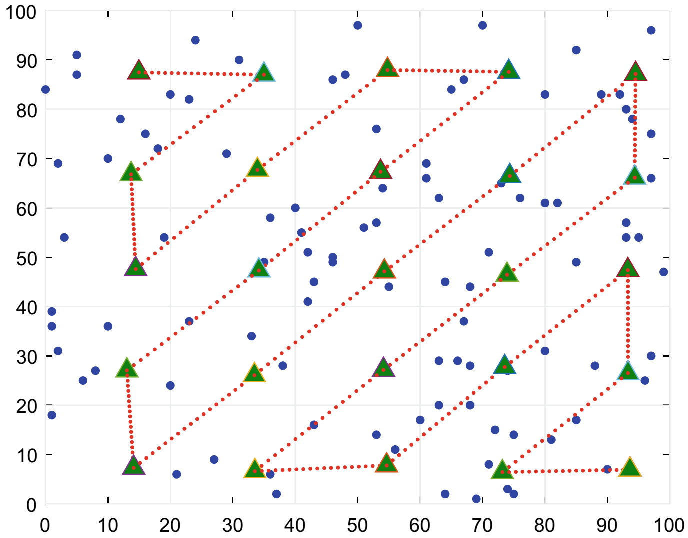


Fig. 4 Trajectory of the mobile beacon node

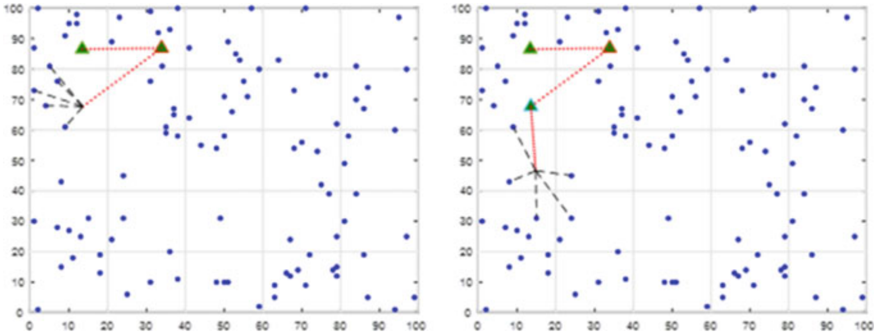


Fig. 5 Location of node positioned by the mobile beacon node

is very poor. The error encountered during the localization is shown in Fig. 6. In that figure, blue colored dots indicate the location of actual node, and green colored dots to the predicted location of sensor node by the PRO algorithm.

4.1 Performance Evaluation

The performance of the proposed model is compared with the PWOA, WCL, CTO, ODR, DSA, TMAL, CLAT, iHRNL, outlier detection, adaptive FPA, ERTC, RRDL, and EW-CSO for localization error whereas, the localization time, localized sensor,

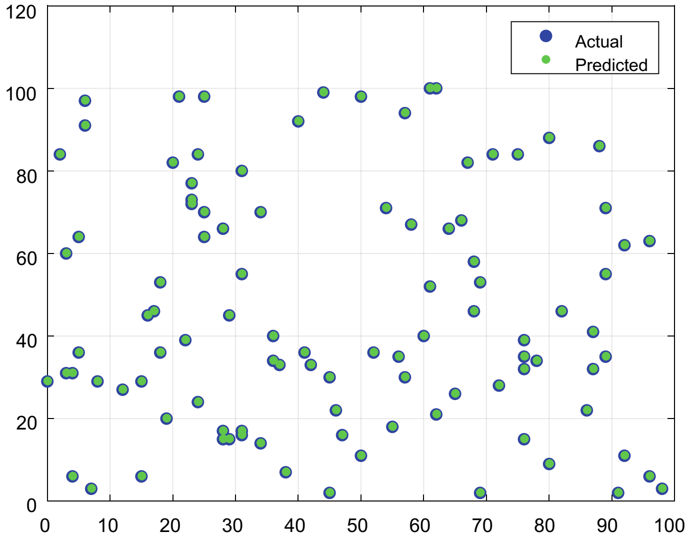


Fig. 6 Actual and estimated location of nodes

convergence rate are compared with the adaptive FPA, ERTC, RRDL, and EW-CSO models.

4.1.1 Localization Error

The distance between the actual placement of sensor node and the perceived location of the node by the used model gives the localization error.

$$\text{Error} = \sum_{\text{sensor}=1}^{S} \left(\left\| e_p - e_a \right\| / R \right) \tag{27}$$

Here, sensor is the number of sensor used and ranges as 1 to S (since for the analysis the sensors used are varied). e_p, e_a are the perceived location and the actual location of the sensor node.

The localization error caused in range-based technique is much lower than the range-free technique is proved in this work. To prove that the proposed work is compared with the existing range-free techniques like PWOA, WCL, CTO, and ODR. The result shows the better performance of the proposed method as in Fig. 7.

Then the multiple beacon node model and the proposed single node model is compared in the Fig. 8. The need for this type of comparison is due to the cost-effective operation. The beacon node operates by the GPS and a power storage unit (battery), the number of beacon node increases the cost also increases. Thus the multiple beacon node system losses the goal of cost-effective system; since the

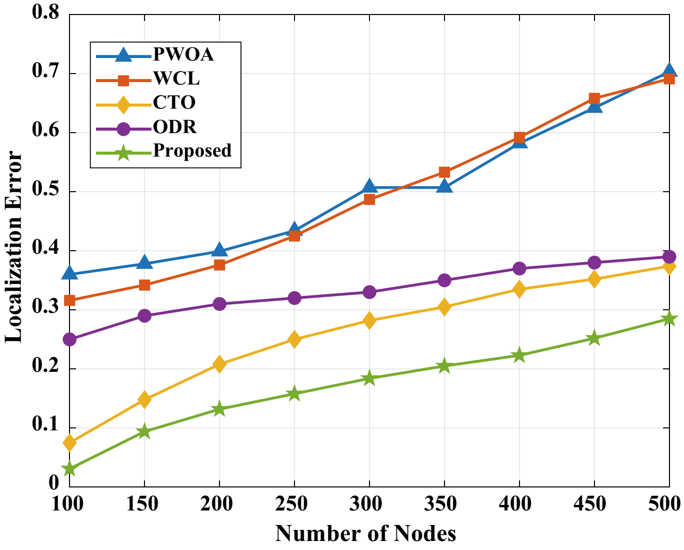


Fig. 7 Localization error compared with range-free techniques

researchers find the localization issue for the cost-efficient operation of sensor node. Hence the model is compared with the multi-node systems like DSA, TMAL, CLAT, iHRNL, and outlier detection. Even though, by employing the multiple beacon nodes the localization error on the model is high compared to the single beacon node model, which violates the need for it.

In Fig. 9, by varying the localization speed the localization error of the proposed model is compared to the model that employs single beacon node such as adaptive FPA, ERTC, RRDL, and EW-CSO. There is only a small variation on the localization error but on real-world problems these ranges are also a significant one. The rest of all the parameters are evaluated and compared with the single beacon node model.

4.1.2 Energy Consumption

Energy consumption is the average energy consumed by the node after several transmissions or several intervals of time. Here, the energy consumption rate is taken for the mobile beacon node. For the mobile beacon node, the energy drainage depends on its movement and also by its communication radius.

$$\text{Energy}\alpha\ (R \cdot \beta_y) \tag{28}$$

Here, the energy is proportional to R communication radius of the beacon node and β_y the parameter for the traversing of node.

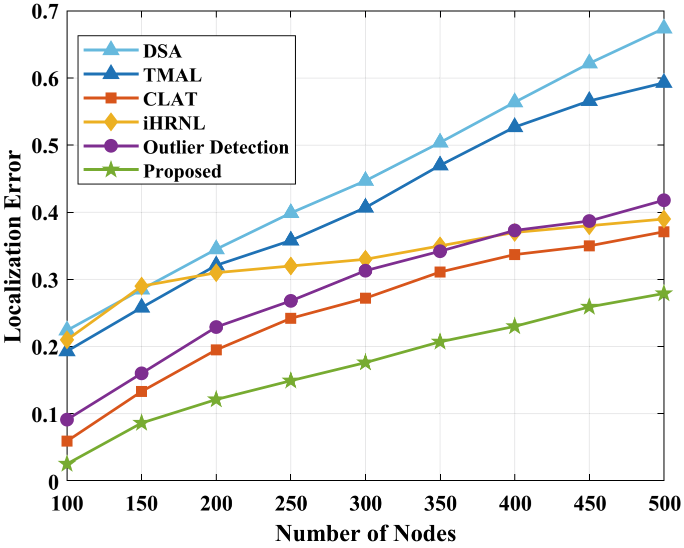


Fig. 8 Localization error compared with multiple beacon node model

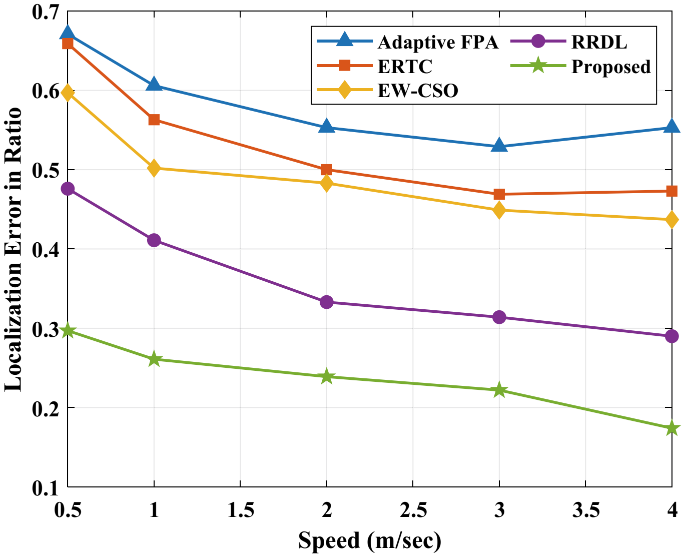


Fig. 9 Localization error compared with single beacon node model

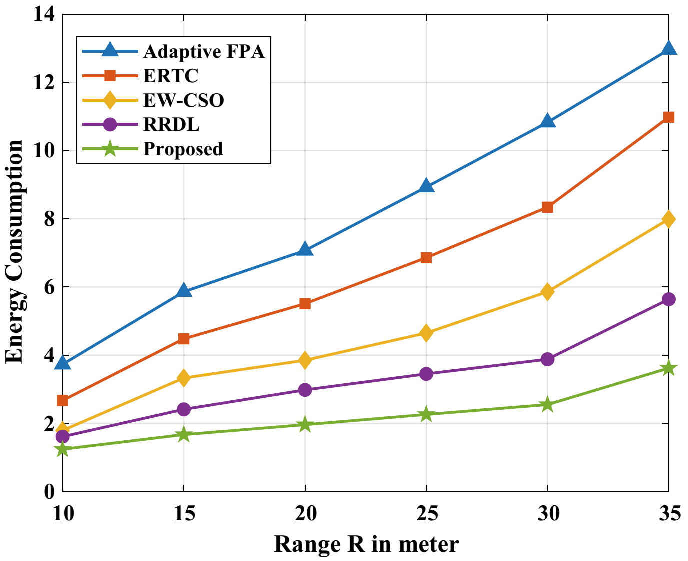


Fig. 10 Energy consumed by the beacon node

Figure 10 shows the energy consumed by the mobile beacon node during the traverse in the network. The energy consumed by varying the communication radius of the beacon node is given here. But for static beacon node model and range-free techniques, the energy consumed will be much lower since the node does not lose energy during traversing. The energy consumption is model is compared with the single beacon node model.

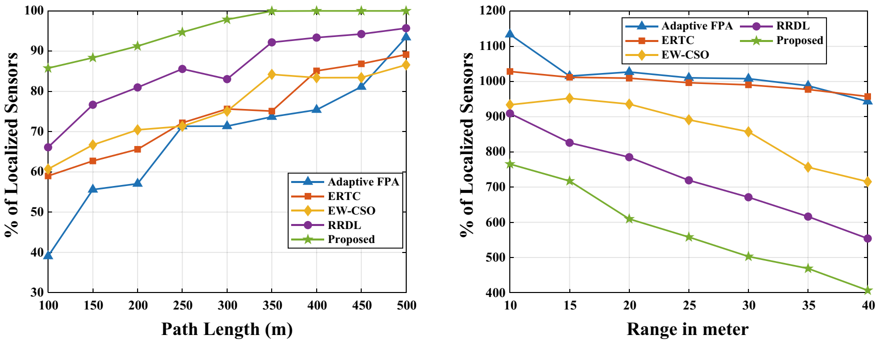
4.1.3 Percentage of Localized Sensors

This performance evaluation term gives the ratio of localized node to the total number of nodes. The localized node is considered along with its accuracy. If the node is localized with high localization error, then the node is not considered as localized node.

$$\%\,\text{locNode} = \frac{\text{locNode}}{S} \tag{29}$$

where, locNode gives the localized node and is the total number of nodes used for the analysis.

The percentage of the localized node concerning path length is given in Fig. 11a. This implies that the number nodes that are localized with high error is very low for the proposed model. Thus making highest number of nodes to get localized. Also



(a) (b)

Fig. 11 Percentage of localized sensors **a** concerning path length **b** concerning range of beacon node

due to the consideration two ranging methods of RSSI and ToA, highest number of nodes are localized as in Fig. 11b.

4.1.4 Accurate Location Information

The accurate estimation gives the number of nodes that are estimated to be error free in localization. The estimation includes the error and the number of nodes participated.

In Fig. 12, the average estimation of number of nodes in the network is given. The estimation is compared with the other techniques like adaptive FPA, ERTC, RRDL, and EW-CSO. By varying the nodes in the field the estimation is shown by the figure. By having the number nodes that are estimated accurately is shown in Fig. 13a whereas, the error on the estimation is given on the Fig. 13b.

4.1.5 Coverage Ratio

Coverage rate gives the average number of nodes S that are covered by the beacon node by its communication radius R.

$$C_{\iota} = {}^{R}\!/_{\mathrm{NS}} \tag{30}$$

where, NS is the network size and R is the communication radius of the beacon node. The coverage rate is measured by setting two cases as varying the network area with 100 s of nodes and varying the sensor nodes on 100 × 100m area.

When the network area is increased for the placement of the 100 node on the same area, the coverage of the sensor node on that case is given in the Fig. 14a. During the large network area, the coverage provided by the proposed model is better than the other model with respect to the total number of nodes considered. In Fig. 14b, the

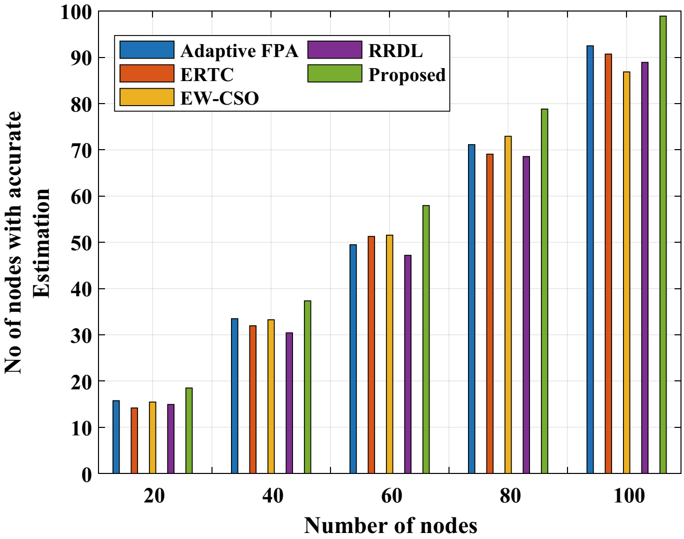


Fig. 12 Accurate localization of nodes

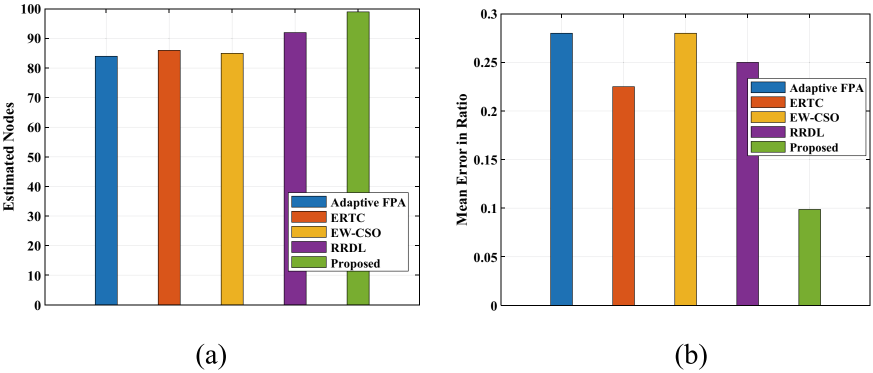


Fig. 13 Accurate localization of nodes **a** estimated nodes **b** mean error

coverage ratio by varying the number of sensor nodes on the network field of 100 × 100 m is shown.

4.1.6 Localization Time

Localization time gives the amount of time taken to localize the node in the network. The time depends on the velocity of the beacon node, by varying the velocity the localization time of the node is calculated.

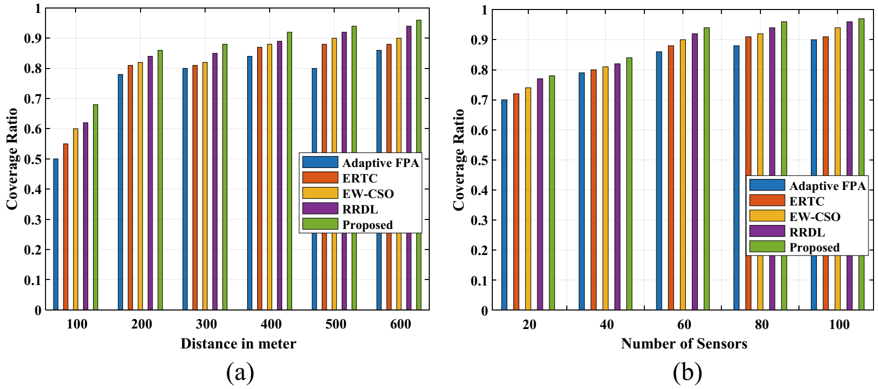


Fig. 14 Coverage ratio **a** with respect to distance **b** with respect to varying sensor node

$$\text{loctime } \alpha \ (\vartheta \cdot r) \tag{31}$$

where, ϑ denotes the velocity of the beacon node and r is the distance between the node and the beacon node.

The comparison the proposed model with the adaptive FPA, ERTC, RRDL, and EW-CSO methods are given in Fig. 15. When the speed of the beacon node increases the localization time is reduced. For the proposed model the localization time is very low.

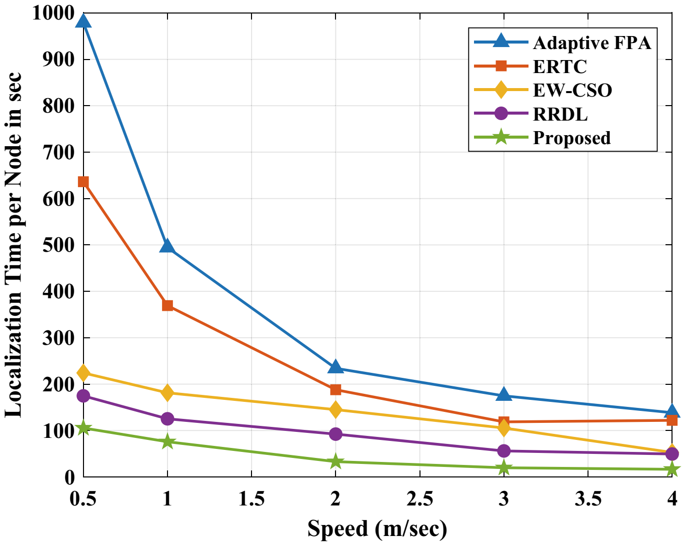


Fig. 15 Comparison of localization time

5 Conclusion

In this research paper, the localization of the sensor node based on ranging model is performed with single beacon node. The starts by considering a trajectory for the mobile beacon node, ranging model, and at last, locating the nodes. The trajectory of the mobile beacon node is known to be a zigzag path so that the nodes on the corner of the network may have high convergence. For the ranging of the nodes at each position of beacon node, the RSSI and ToA of the nodes are found so that the maximum number of nodes would get localized. Then the coordinates of the nodes were set by the proposed PRO algorithm and its performance was analyzed by MATLAB platform. The model have yielded very low error rate of 0.002% for the 100 nodes on the network having 92 nodes as perfectly analyzed with 0.9 coverage ratio.

References

1. Muduli L, Mishra DP, Jana PK (2018) Application of wireless sensor network for environmental monitoring in underground coal mines: a systematic review. J Netw Comput Appl 106:48–67
2. Priyadarshini R, Raj N, Sivakumar (2019) Enhancing coverage and connectivity using energy prediction method in underwater acoustic WSN. J Ambient Intell Humanized Comput, pp 1–10
3. Kim DS, Tran-Dang H (2019) Wireless sensor networks for industrial applications. In: Industrial sensors and controls in communication networks, pp 127–140. Springer
4. Elhoseny M, Hassanien AE (2019) Hierarchical and clustering WSN models: their requirements for complex applications. In: Dynamic wireless sensor networks, pp 53–71. Springer
5. Verma SN, Sharma AK (2018) Design of a novel routing architecture for harsh environment monitoring in heterogeneous WSN. IET Wireless Sens Syst 8(6):284–294
6. Mehra P, Singh M, Doja B, Alam (2019) Stability enhancement in LEACH (SE-LEACH) for homogeneous WSN. EAI Endorsed Trans Scalable Inf Syst 6(20)
7. Numan M, Fazlisubhan W, Khan S, Sajjadhaider G, Reddy A, Mamounalazab (2020) A systematic review on clone node detection in static wireless sensor net- works. IEEE Access 8:65450–65461
8. Elhoseny M, Hassanien AE (2019) Expand mobile WSN coverage in harsh environments. In: Dynamic wireless sensor networks. pp 29–52. Springer
9. Srivastava S, Singh M, Gupta S (2018) Wireless sensor network: a survey. In: 2018 international conference on automation and computational engineering (ICACE) pp 159–163 (2018)
10. Chelouah L, Louizabouallouche-Medjkoune F (2018) Localization protocols for mobile wireless sensor networks: a survey. Comput Electr Eng 71:733–751
11. Naureen A, Zhang N, Furber S, Shi Q (2020) A GPS-less localization and mobility modelling (LMM) system for wildlife tracking. IEEE Access 8:102709–102732
12. Kannadasan K, Edla DR, Chowdarykongara M, Venkatanareshbabukuppili (2019) M-curves path planning model for mobile anchor node and localization of sensor nodes using dolphin swarm algorithm. Wireless Netw, pp 1–15
13. Tuba E, Tuba M, Beko M (2018) Two stage wireless sensor node localization using firefly algorithm. In: Smart trends in systems, security and sustainability. pp 113–120. Springer
14. Shah S, Bilal C, Zhe F, Yin I, Khan S, Begum M, Faheem F (2018) Khan: 3D weighted centroid algorithm & RSSI ranging model strategy for node localization in WSN based on smart devices. Sustain Cities Soc 39:298–308

15. Alshamaa D, Mourad-Chehade F, Honeine P (2019) Decentralized kernel-based localization in wireless sensor networks using belief functions. IEEE Sens J 19(11):4149–4159
16. Qiao G, Zhao C, Zhou F, Ahmed N (2019) Distributed localization based on signal propagation loss for underwater sensor networks. IEEE Access 7:112985–112995
17. Wang T, Wang X, Shi W, Zhao Z (2020) Target localization and tracking based on improved Bayesian enhanced least-squares algorithm in wireless sensor networks. Comput Netw 167:106968–106968
18. Achroufene A, Yacineamirat, Abdelghanichibani (2018) RSS-based indoor localization using belief function theory. IEEE Trans Automation Sci Eng **16**(3), 1163–1180
19. Wu H, Mei X, Chen X, Li J, Wang J (2018) Prasantmohapatra: a novel cooperative localization algorithm using enhanced particle filter technique in maritime search and rescue wireless sensor network. ISA Trans 78:39–46
20. Shahra E, Tarekrahilsheltami SEM (2020) A comparative study of range-free and range-based localization protocols for wireless sensor network: using cooja simulator. Sens Technol: Concepts Methodologies Tools Appl, pp 1522–1537
21. Sneha V, Nagarajan M (2020) Localization in wireless sensor networks: a review. Cybern Inf Technol 20(4):3–26
22. Bhat SJ, Santhosh KV (2020) Is localization of wireless sensor networks in irregular fields a challenge? Wirel Pers Commun 114:2017–2042
23. Zhang S, Yan S, Hu W, Wang J (2015) Kehuaguo: a component-based localization algorithm for sparse sensor networks combining angle and distance information. KSII Trans Internet Inf Syst 9(3):1014–1034
24. Yu XW, Huang LP, Yong LIU, Hao YU, Ying LI (2021) Convex localization algorithm based on time difference of arrival for WSN in uranium tailings radioactive contamination. Wirel Pers Commun, pp 1–17
25. Singh P, Mittal N (2021) An efficient localization approach to locate sensor nodes in 3D wireless sensor networks using adaptive flower pollination algorithm. Wirel Netw, pp 1–16
26. Liu X, Yin J, Zhang S, Ding B, Guo S, Wang K (2018) Range-based localization for sparse 3-D sensor networks. IEEE Internet Things J 6(1):753–764
27. Huang H, Miao W, Min G, Huang C, Zhang X, Wang C (2020) Resilient range-based d-dimensional localization for mobile sensor networks. IEEE/ACM Trans Net-working 28(5):2037–2050

Design of Mobile Application for Farmers

S. Gayathri Devi, S. Chandia, and K. Saraswathi

Abstract India is a global agricultural powerhouse, with farmers and other related workers serving as its backbone. The agricultural environment, like many other industries, is plagued by long-standing issues and unanticipated challenges that must be addressed. This study has taken a look at some of the most pressing concerns that farmers have. We identified problems in conveying correct and timely information about the crops to grow, weather, pesticides, fertilisers, various government schemes on agriculture, disease detection methods. The digital form of information has reached the farmers quickly. Nowadays, normally they have possessed smartphones so that information through this device is a right way to reach them fastly. Although several agriculture-based mobile applications are already available, the application in the regional language will benefit the farmers. In Tamil Nadu, the Uzhavan mobile application is developed and released by the state government in the Tamil language. This application has its disadvantages. When we communicate with the nearby farmers, they gave their needs. They have collected through a survey form. From that detail, we present the difficulties among the farmers in the Coimbatore district. After this identification stage, we have tried to model an Agri-Tech product in Tamil and deliver the information on their requirements. Finally, the benefits and future scope of the proposed model have been included in this paper.

Keywords Difficulties on agriculture · Mobile application · Agri-Tech product

S. G. Devi (✉) · S. Chandia
Department of Computing, Coimbatore Institute of Technology, Coimbatore, Tamil Nadu 641014, India
e-mail: sgayathridevi@cit.edu.in

K. Saraswathi
Kongu Engineering College, Erode, Tamil Nadu, India

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_11

1 Introduction

Indian history confirms its agricultural efficiency, favourable climatic circumstances, and abundant natural resources. India (which cultivates wheat, rice, and cotton on large swaths of its land) is also a major exporter of spices, pulses, and milk on the worldwide market. India is the world's second-largest producer of agricultural products, with $375.61 billion in production. India produces 7.39% of the world's total agricultural output. India lags well behind China, which has a $991 billion GDP in agriculture. The agricultural industry contributes significantly more to the Indian economy than the global average (6.4%). The participation of the industry and services sectors is lower than the global average of 30% for the industrial sector and 63% for the services sector [1]. In India, the agricultural sector employs around 52% of the entire workforce. India's agriculture sector accounts for over 43% of the country's total land area. Agriculture is the backbone of an economy that provides people with fundamental foodstuffs and, more recently, raw materials for industrialization [2]. As a result, agriculture's contribution to economic development can be summarised as follows:

- Contribution to national income
- Source of food
- Prerequisite for raw material
- Infrastructure creation
- Aid in recovering from economic depression

Coimbatore is located in the Kongu Nadu area of Tamil Nadu, in the western section of the state. The district is bordered on the west by the Palakkad district of Kerala, on the north by the Nilgiris district, on the east by Erode district, on the south by Idukki district of Kerala, and on the southeast by Dindigul district. The district of Coimbatore is located in Tamil Nadu's northern section, between 11°15' north latitude and 77°19' east longitude, with an average sea level (MSL) of 432-12 m. The main crops grown in the Aliyar and Palar sub-basins of this region are coconut, banana, maize, sugarcane, and tapioca. There are three types of soils in these villages: black loam, red loam, and red sandy soils. Red loam, red sand, and black loamy soils account for 15%, 10%, and 75% of the total soils, respectively [3]. The district of Coimbatore covers a total geographic area of 367,097 hectares, with a net cultivated area of roughly 165,260 hectares. Coconut is the most important plantation crop, which is grown on an area of roughly 85,831 hectares. Millets, pulses, oilseeds, cotton, and sugarcane are among the other crops grown.

The importance of information is universal and unquestionable. Independent of the specific benefits to farmers, providing information to others who do not have it and who are in desperate need is a valuable service. Both value additions in the Indian agriculture sector and value-added services in the mobile phone industry are in desperate need of attention and both have the potential to improve the lives of farmers while also creating value for a variety of stakeholders, including mobile service providers and content aggregators [4].

For agriculture, fast access to the relevant information and proper application of that information is critical. Farmers can profit from ICT-based initiatives for the dissemination of information, the transfer of technology, the procurement of inputs, and the sale of outputs. Farmers may adopt appropriate agricultural practices, make better input choices, and plan their crops better with timely information and practical solutions to agricultural challenges [5].

The rest of this paper has been arranged as follows. Section 2 depicts the common issues that Indian farmers encounter. Section 3 explains the information and communication technology in the agriculture industry which includes the usage of mobile phones among farmers and analysis of existing applications. Section 4 identifies the difficulties faced by farmers through the survey results. It also describes survey questions about farming. Section 5 depicts the features, benefits, and architecture of the proposed system. It also shows the Adobe XD prototype for the proposed application. Finally, Sect. 6 concludes this study with future work.

2 The Most Common Issues that Indian Farmers Encounter

2.1 Inadequate Water Supply

In India, there is more than enough water to irrigate all cultivated regions; the challenge is that we must still discover inexpensive and appropriate ways to make use of such vast water reserves. Farmers either do not obtain enough water or do not receive it on time for a variety of reasons; many farmers rely on rainwater for irrigation.

2.2 A Reduction in the Use of Modern Farming Equipment

Farmers still use primitive agriculture methods in most places, preferring a conventional plough and other native accessories. Despite the availability of effective equipment and machinery, contemporary equipment is rarely used, because most farmers do not have lands large enough to accommodate advanced instruments and heavy machinery.

2.3 An Over-Reliance on Traditional Crops

Rice and wheat have been grown by Indian farmers in different places for ages. Inadequate water supply excessive production of these two grains frequently results in storage issues, sales issues, and a lack of other farm products. According to the

US Department of Agriculture, “India is on track for a fourth record wheat harvest and near-record rice production in 2020–21.” Many farmers relying solely on these traditional crops suggests a lack of a national agriculture strategy.

2.4 Insufficient Storage Facilities

Storage facilities in remote areas are either limited or non-existent. Farmers in such a position frequently have no choice except to sell their products as soon as it is ready, at market prices that are sometimes quite low. They are a long way from having a legal source of income.

2.5 Issues with Transportation

In India’s agriculture sector, a lack of affordable, effective transportation is a major issue; small farmers still rely on bullock carts to deliver their produce. Furthermore, lakhs of villages are linked to highways and market centres via temporary (kutcha) roads that become muddy and impassable during rainy seasons. As a result, farmers are unable to get their produce to the central market and are forced to sell it at a cheap price in the local market.

2.6 High-Interest Rates

Thousands of farmers commit suicide each year as a result of debt (having other indirect causes interlinked). Unreasonably high-interest rates should be deemed unlawful, and the government should act swiftly, harshly, and appropriately against greedy money lenders. Another issue is that small and marginal farmers must go through lengthy procedures (of which they are unaware) to obtain institutional finance.

2.7 Small Farmers Have yet to Benefit from Government Programme

The government implemented an agricultural debt waiver and debt-relief programme in 2008, benefiting approximately 36 million farmers. Direct agricultural loans to distressed farmers were also included in the scheme. However, the majority of such welfare programmes and subsidies declared by both the federal and state governments have yet to reach poor farmers, whereas large/wealthy landowners have reaped the benefits.

3 Information and Communications Technology in Agriculture

Farmers have expressed a desire for fast access to information on technology, monsoon patterns, and market opportunities. In most cases, regular extension services do not reach the farmer at the appropriate time and location. Modern information and communication technology (ICT) opens up new avenues for rural households, farmers, fishermen, and women to bridge the information divide [6]. Agriculture has new and important obstacles and challenges every day. Using information and communication technology (ICT) can help farmers produce crops more efficiently. Simply said, information and communication technology (ICT) is a collection of technologies that help with data storage, processing, communication, and delivery [7].Traditional communication aids (e.g. telephones, televisions), the Internet, and mobile applications, as well as Big Data analytics and information systems, cloud computing, the Internet of Things, remote sensing and drones, blockchain, and artificial intelligence, are all examples of information and communication technologies (ICTs) [8]. In practical, every area of agriculture, communication, information exchange, transactions, and knowledge transfer is essential. As a result, agricultural and food chain digitization are high on the political agenda [9].

3.1 Usage of Mobile Phones among Farmers

Many farmers rely solely on their cell phones. Farmers, on the other hand, prefer to get messages/advisories in multimedia style these days. As a result, they are equipped with smartphones that allow them to receive Hi-tech alerts. The department of agriculture should take the necessary efforts to give farmers low-cost smartphones. They should also have access to the Internet at a reduced cost. The service provider should use a consistent frequency and time for sending messages to ensure that the farmer subscribers are kept informed. The baseline survey should be conducted by the service provider to learn about the needs of farmers, their cropping patterns, and the types of mobile phones they use. The development of relevant material, improved penetration, and reduced inequality will all be beneficial. Farmers' requirements should be assessed regularly at the village level to meet the diverse and growing demands of farmers [10].

3.2 Analysis of Existing Applications for Farmers

Table 1 lists numerous agricultural apps that are already accessible in various languages, along with their purpose, modules, supported languages, and shortcomings. Government schemes announcement, crop management, weather forecasting,

Table 1 Comparative analysis of various mobile applications for agriculture industry in India

S. No.	App name	URL	A	B	C	D	E	F	G	Supporting language(s)
1	IFFCO—Kisan	http://iffcokisan.com/	✓	x	✓	✓	x	x	x	English
2	Agri central	https://www.androidfreeware.net/download-com-globalagricentral.html#:~:text=Download%20AgriCentral%20APK%20for%20Android%20and%20install.%201,5%3A%20Launch%20the%20app%20and%20enjoy.%20More%20items	x	✓	✓	x	x	✓	x	English, Hindi, Marati, Telugu, Kanadam
3	Plantix	https://plantix.net/en/	x	✓	✓	x	x	x	x	English, Hindi, Marati, Telugu, Kanadam
4	Bharath Agri	https://www.bharatagri.com/	x	✓	x	x	✓	✓	x	Hindi, English
5	Krishi Network	https://krishinetwork.com/	✓	x	✓	x	x	x	x	Hindi
6	Farm Key	https://play.google.com/store/apps/details?id=agriculture.farmers.app.farmkey&hl=en_US&gl=US	x	x	✓	x	x	✓	x	English
7	Agri app	https://www.agriapp.co.in/	x	x	x	x	✓	✓	✓	Supported in 11 languages
8	Agri Science Krishi	http://agriscienceindia.com/	✓	x	x	✓	x	✓	x	Gujarati
9	Crop Bee	https://www.cropbee.in/	x	✓	x	✓	x	✓	✓	Telugu, English
10	Agriplex	https://agriplexindia.com/	✓	x	x	x	x	x	✓	English
11	Krishify Kissan App	https://www.krishify.com/	✓	x	✓	x	x	x	x	Hindi
12	Crop bag	https://play.google.com/store/apps/details?id=com.cropbag.farmer&hl=en_IN&gl=US	x	✓		x	x	x	x	English
13	India Krushi Kendra	https://www.krushikendra.com/	x	x	x	x	x	x	✓	English

(continued)

Table 1 (continued)

S. No.	App name	URL	A	B	C	D	E	F	G	Supporting language(s)
14	Kheti point	https://www.khetipoint.in/	x	✓	✓	x	x	x	x	Hindi, Urdu
15	Mandi central	https://apps.apple.com/in/app/mandi-central/id1434800656	x	x	x	✓	x	✓	x	English, Hindi
16	Big Haat	https://www.bighaat.com/	x	✓	x	x	x	x	x	English, Hindi, Telugu
17	Krushikendra Online Agriculture Megastore	https://www.krushikendra.com/	x	x	x	x	x	x	✓	English
18	Kheti Gaadi	https://khetigaadi.com/	x	x	x	x	x	✓	x	English
19	Uzhavan	https://www.tnagrisnet.tn.gov.in/people_app/	✓	✓	✓	x	✓	✓	x	Tamil
20	Vivasayam	https://www.envivasayam.com/category/%e0%ae%b5%e0%af%80%e0%ae%9f%e0%af%8d%e0%ae%9f%e0%af%81%e0%ae%a4%e0%af%8d%e0%ae%a4%e0%af%8b%e0%ae%9f%e0%af%8d%e0%ae%9f%e0%ae%ae%e0%af%8d/	x	x	✓	x	✓	✓	✓	Tamil
21	Kamatan	https://www.kamatan.in/	x	x	✓	✓	✓	✓	x	Hindi
22	Agri Bolo	https://www.agribolo.com/	✓	x	✓	✓	x	✓	x	Hindi and English
23	Agrowon	https://www.agrowon.com/	x	x	x	x	x	✓	x	Hindi and English
24	De Haat Kisan	https://www.androidfreeware.net/download-app-intspvt-com-farmer.html	x	✓	x	✓	x	x	x	Hindi
25	Agro Infomart	https://www.agroinfomart.com/	x	x	x	x	x	✓	x	English

(continued)

Table 1 (continued)

S. No.	App name	URL	A	B	C	D	E	F	G	Supporting language(s)
26	Jai kisaan	http://jaikisaan.in/login	x	✓	✓	x	✓	✓	x	English, Marathi Kannada, Tamil, Telugu, Hindi, Gujarati, Oriya, Malayalam
27	First Farmer	https://apkpure.com/first-farmers-business-mobile/com.mfoundry.mb.android.mb_beb581	x	x	x	✓	x	✓	x	English, Gujarati, Hindi
28	Kisaan	https://apkpure.com/mp-kisan-app/in.gov.mapit.kisanapp#:~:text=%20The%20description%20of%20MP%20Kisan%20App%20App,Link%20own%20Khatas%20through%20Aadhar%20Number.%20More%20	x	✓	x	x	x	✓	x	Hindi, English, Urudu and Tamil
29	Farmrise	https://www.appbrain.com/app/farmrise-mandi-prices-weather-update-agronomy/com.climate.farmrise#:~:text=%20Description%20%201%20FarmRise%20app%20is%20an,sourced%20from%20media%20and%20public%20libraries%20More%20	x	x	✓	✓	✓	✓	x	English, Hindi, Marathi, Kanadam

A Govt. schemes announcement, *B* Crop management, *C* Weather forecasting, *D* Market demands display, *E* Education videos for cultivation and organic farming, *F* Announcement of new Agri-Tech products, *G* Establishing market and farmers connection market demands display, education videos for cultivation and organic farming, announcement of new Agri-Tech products, establishing market, and farmers connection are some of the application functions. With all of the characteristics combined, none of the following programmes addresses the farmers' overall requirement in their native language [11].

According to Don Richardson [14], the use of ICTs in support of agricultural and rural development applications can be divided into five categories. Agricultural producers' economic development, community development, research and education, small and medium firm development, and media networks are among them. Mobile apps can digitalize the planet, thanks to the recent advances in ICT and their application in agriculture. Mobile agriculture applications are those that focus on the needs of the agricultural sector and its stakeholders, such as farmers, input merchants, and collaboration [12]. The Tamil Nadu government has introduced a new smartphone application called Uzhavan for spreading agriculture-related information to farmers. The limitations that agricultural extension officers confront when utilising the Uzhavan app are discussed in this research. Data was collected using a well-structured interview schedule. One hundred percent of extension officers used the Uzhavan app on their smartphones. Extension officers commonly used the Uzhavan app's market price and seed stock features to provide advice to farmers. The main issue that extension officers had while using the Uzhavan app was that the information was not kept up to date [13].

4 Difficulties Faced by Farmers

4.1 Questionnaire for Survey

The purpose of the questionnaire is to learn about the habits of farmers during the agricultural process. The questions are divided into groups based on the crop's life cycle. The inquiry also covers product storage and delivery to the market. There are 35 questions total in the questionnaire, which is organised under various areas. Due to the pandemic situation, a considerable number of agriculturists responded. Farmers are prompted to fill out general information at first. The crop selection group is responsible for gathering information such as organic awareness, market needs, and crop selection. The field preparation obtains information on testing centre awareness. The seeding section is for knowing the awareness about natural seeds and quality. The next session is irrigation. This deals with the details about water sources and availability. The sprouting stage asks about the climate condition, pests, disease, and weed control mechanisms. In the harvesting stage, how the products are sold and any help from government sectors are the questions. Finally, the storage and marketing stage asked about the storage facilities, market price, and help from government organisations. Thus, the questionnaire is designed to collect information about cultivation, modern techniques used, government support for the farmers, and how they get awareness about farming.

4.2 Analysis of Survey's Responses

The questionnaire was distributed as a Google form in this epidemic context. This form might be auto-translated into Tamil so that many farmers can comprehend and respond to the questions. They were scattered throughout the Coimbatore district. The majority of respondents cited a lack of irrigation infrastructure, a lack of awareness of natural farming and market demands, price fluctuations, and the participation of middlemen in marketing as issues. In terms of proposals, raising knowledge of existing schemes, weather conditions, market crop demand, and ICT-based technology transfer in their regional language would help.

4.3 Discussion

A survey of Coimbatore farmers' agricultural status was conducted, however, only a few people replied due to the coronavirus pandemic. The following difficulties were discovered as a result of the survey. Farmers in and around Coimbatore confront a variety of issues. The following are the numerous challenges faced by farmers in and around Coimbatore, as reported by a survey conducted among farmers:

- The unawareness of soil type and cultivation methods
- No direct communication with boards that are run by state and central government
- Unawareness of side effects of chemical fertilisers and pesticides
- A significant disconnect between farmers and customers
- A lack of assistance from government agencies
- The inability of various government programmes to reach the intended audience
- No clear idea on market demands
- Storage and marketing-related issues
- Limited access to market data

The following charts have been taken from the responses to the survey. Figure 1 shows that 68% of the farmers have smartphones so that they can access mobile apps.

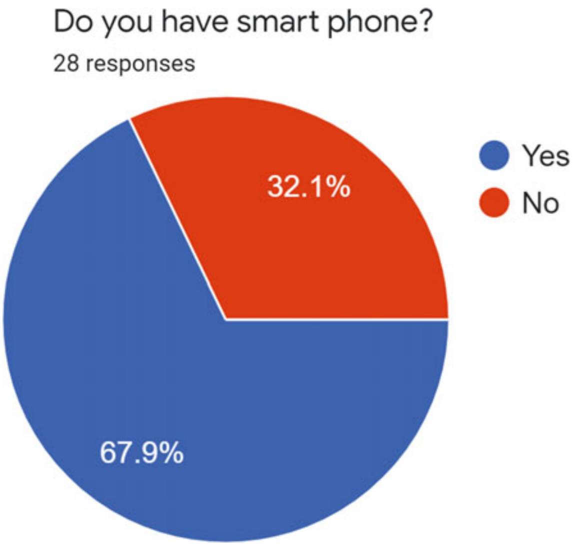


Fig. 1 Availability of smartphone

Figure 2 reveals that 72% of cultivation methods adopted by the farmers are mixed, i.e. a mixture of modern and traditional methods.

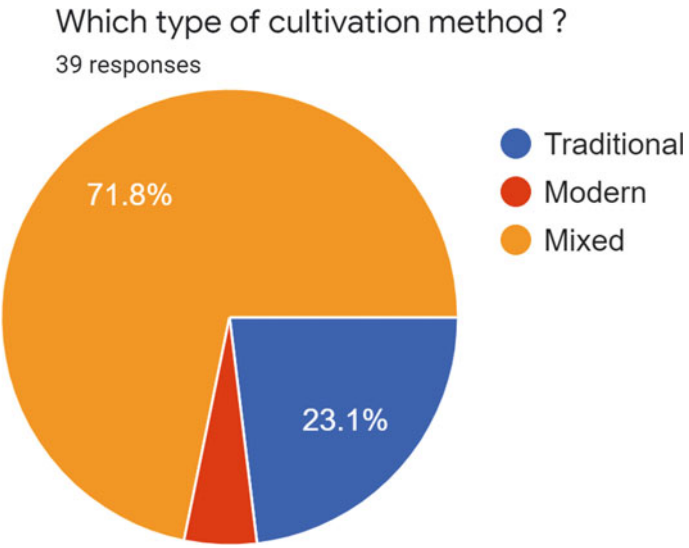


Fig. 2 Type of cultivation method

Figure 3 shows that 85% of farmers want to know about crops cultivated by neighbours. This helps to meet the market demands.

Figure 4 shows that 55% of farmers are not sure about the quality of seeds they purchased and in Fig. 5, 58% of them do not have interaction with the buyers.

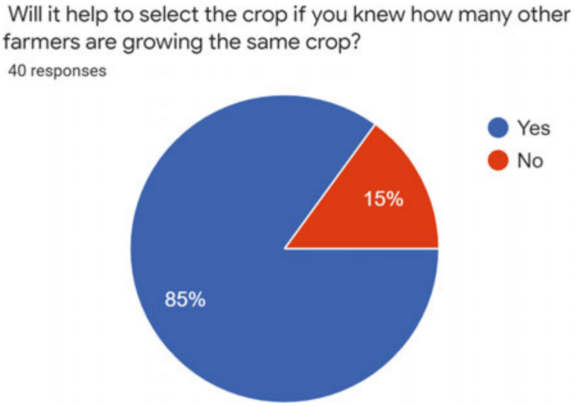


Fig. 3 Knowing nearby farmers' crop

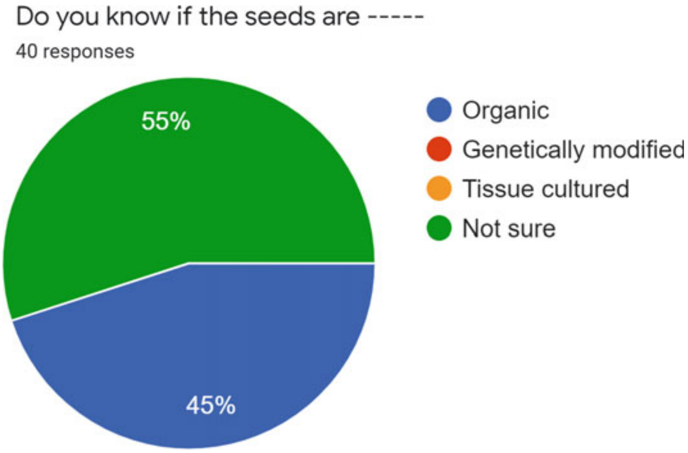


Fig. 4 Quality of seeds

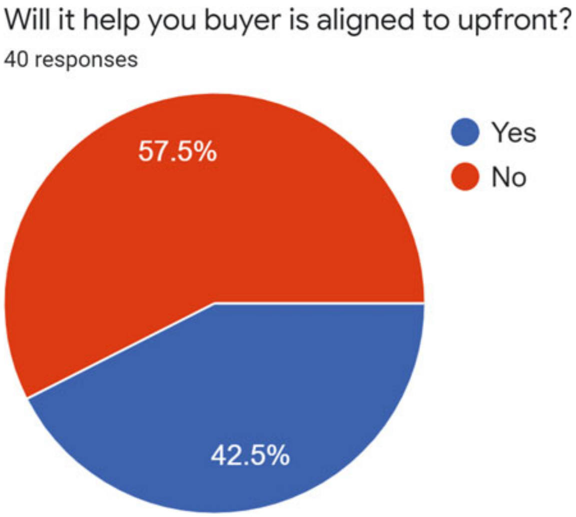


Fig. 5 Buyer interaction

Figure 6 describes the government involvement in the marketing of crops. About 61% of responses showcase that there is no involvement in government agencies. The aforementioned issues paved the way for the proposed work of digital farming, which entails designing and developing an agriculture application that provides various information to farmers online, such as agriculture advice, various market prices, new government schemes related to agriculture, suggestions for choosing the appropriate crop variety for their soil type, and weather location, which aids in the imposition of the proposed work of digital farming. The project also intends to assist farmers in learning about the latest agricultural trends.

Community development can also benefit from the usage of modern information technologies. Information technology may be utilised for rural communication to enhance participation, disseminate information, and exchange knowledge and skills for community development when it is systematically applied and suited to the requirements of rural communities [15].

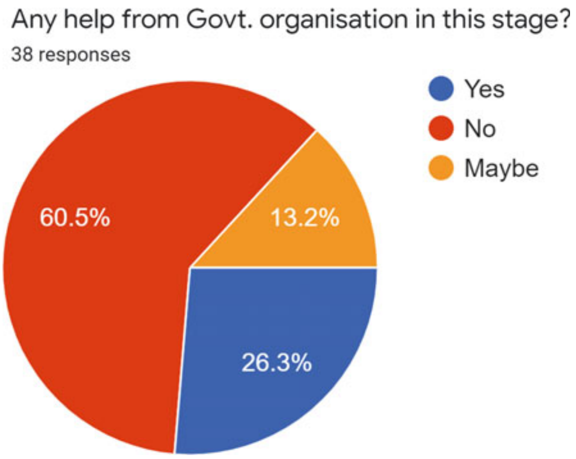


Fig. 6 Government involvement in marketing

5 Proposed Model

The main motivations to create the proposed product are listed below:

- Unawareness of the cultivation demands in the markets among farmers
- Difficulties in the usage of a high-level application already existing in other languages

5.1 Features

Features of the proposed system:

- Application in the Tamil language
- A bridge between market and farmers through displaying nearest market's demands
- Weather forecasting
- Identifying, translating, publishing, and notifying relevant details about government schemes to the target farmers based on the farmer profile
- Creating awareness about modern and organic farming through education videos

5.2 Benefits

Table 2 lists the benefits of the proposed system against the various types of customers.

Table 2 Customer segments versus benefits

Customer segments	Benefits
Farmers	Authentic information, insights, decision support, integration with markets, govt. corporates, and agri-stakeholders
Farmers-procurement agents	Information of crop availability, reducing procurement friction
Government (Local)	Budgeting, forecasting, policymaking, real-time crop blueprint
Government (Central)	Forecasting, budgeting, policymaking, budgeting, forecasting, policymaking, real-time crop blueprint
Universities	Real-time picture

5.3 Architecture Diagram for Proposed Work

Figure 7 shows the architecture diagram for the proposed application. The user registration and display of appropriate information based on user's requirement have been placed in presentation layer. The application layer has the business logic and it has the user manager, weather forecaster, profile matcher, and external source manager. The data layer has the data source for the information. Finally, the data layer has a link with external third party layer.

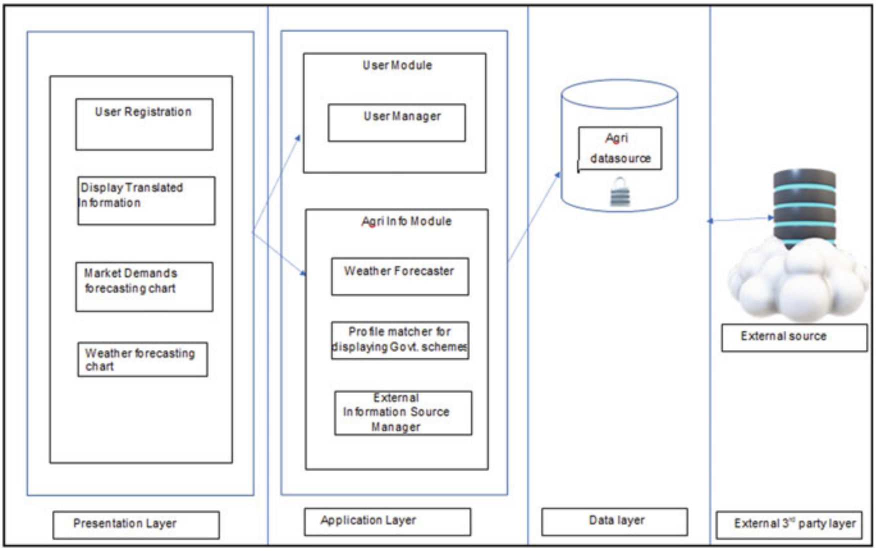


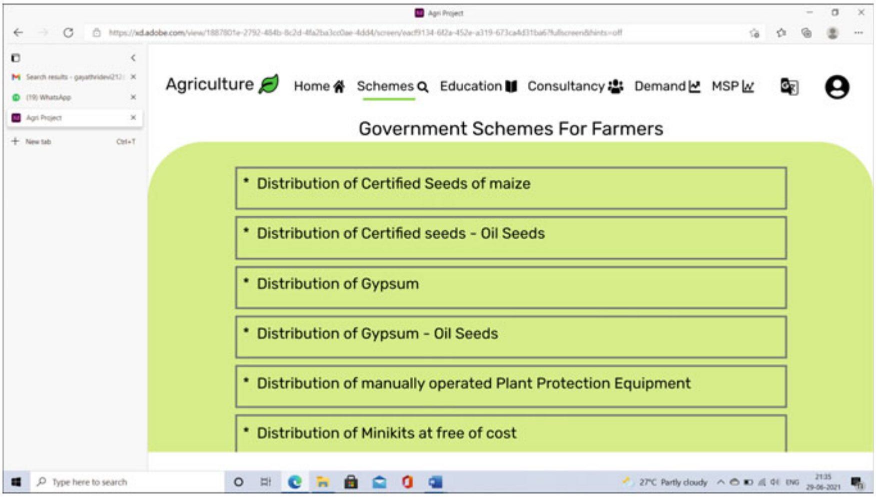
Fig. 7 Architecture diagram for proposed mobile application

5.4 Prototype of Proposed Mobile Application

A high-level prototype has adopted for this work. These screens are shown to farmers for getting feedback from them. These are designed using Adobe XD. Some of the screens are presented here.

The new schemes announcements from state and central government related to the agricultural industry are displayed in Fig. 8. To educate the agriculturists, education/awareness videos are uploaded in the proposed system as in Fig. 9.

The link between farmers and experts like members in agriculture universities, officers allotted for their region can be established through the screen which is shown in Fig. 10. Figure 11 lists the product-wise demands in the market. This will help the farmers to cultivate their crops according to their demands.



Government Schemes For Farmers

- Distribution of Certified Seeds of maize
- Distribution of Certified seeds - Oil Seeds
- Distribution of Gypsum
- Distribution of Gypsum - Oil Seeds
- Distribution of manually operated Plant Protection Equipment
- Distribution of Minikits at free of cost

Fig. 8 Screen to list Government schemes

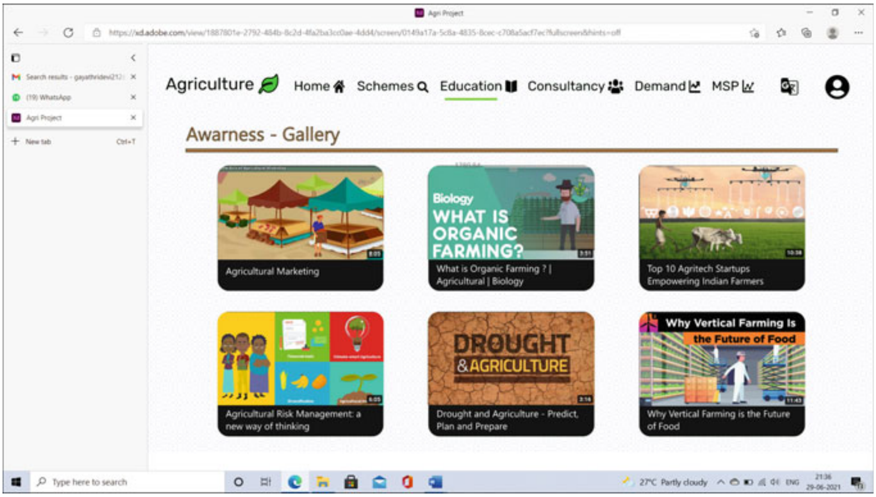
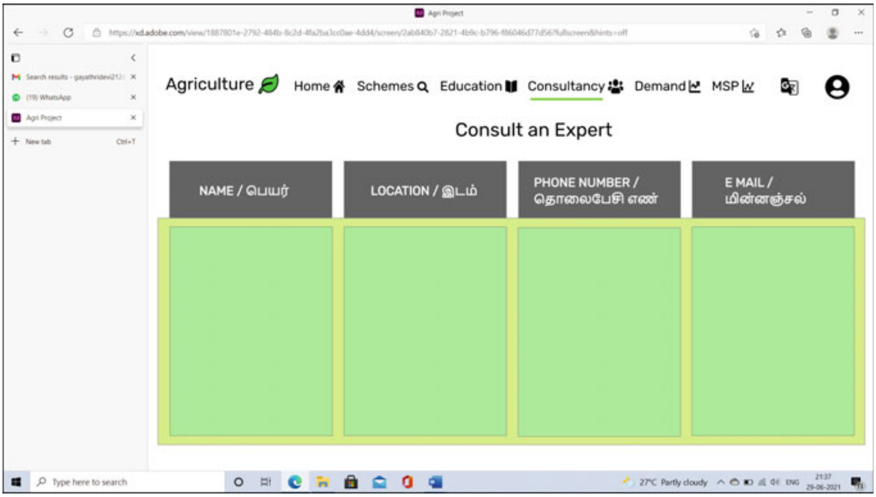

Agri Project
Agriculture
Home
Schemes
Education
Consultancy
Demand
MSP
Awarness - Gallery
Agricultural Marketing
What is Organic Farming ? | Agricultural | Biology
Top 10 Agritech Startups Empowering Indian Farmers
Agricultural Risk Management: a new way of thinking
Drought and Agriculture - Predict, Plan and Prepare
Why Vertical Farming is the Future of Food
Type here to search
27°C Partly cloudy
21:36
29-06-2021

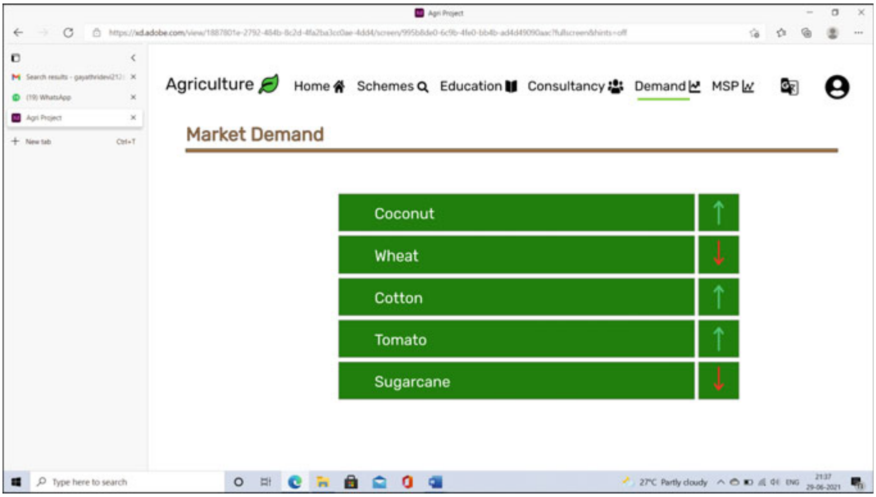
Fig. 9 Screen to display the awareness videos



Consult an Expert

NAME / பெயர்	LOCATION / இடம்	PHONE NUMBER / தொலைபேசி எண்	E MAIL / மின்னஞ்சல்

Fig. 10 Screen for expert interaction



Market Demand

Coconut	↑
Wheat	↓
Cotton	↑
Tomato	↑
Sugarcane	↓

Fig. 11 Screen for Displaying market demands as per product

6 Conclusion

The agri-sector will undoubtedly grow if we encourage young people to pursue farming and related careers. They already have a foundation of institutional knowledge and education, and they can learn and expand quickly. For example, practically, all of them have smartphones, and they may perform effectively on farms by using a sophisticated agriculture app. Furthermore, bringing current technologies and providing innovative equipment to small farmers will aid in increasing efficiency, output, and quality. Digital farming can play a vital role in transmitting appropriate information to farmers which would benefit them in identifying irrigation facilities, markets' demands, price fluctuations, suggestions, existing government schemes, weather conditions, and ICT-based transfer of technology. Most of the agricultural applications discussed in this paper focus on a specific feature and provide functionalities on the same. Also, the barrier of the language in those applications is the greatest issue. All the above issues drive the design and development of the proposed Agri-Tech product that focuses on combining all the features and reaching farmers in their regional language. Finally, we again address a range of limitations of our work. First, we make use of survey data from our survey conducted among the farmers in and around the Coimbatore district. Second, a majority of farmers in our experiment have smartphones and also they are accustomed to use ICT-based applications. Third, our outcome is the prototype design of the proposed application. In ongoing work, we will extend this survey to other districts of Tamil Nadu and enhance the proposed application. Based on those data, we will develop a mobile application for farmers in the Tamil language.

References

1. India GDP sector-wise 2020 - StatisticsTimes.com. (2021, June 17). Statisticstimes.com. https://statisticstimes.com/economy/country/india-gdp-sectorwise.php
2. The State of Food and Agriculture, 1996. (n.d.). www.fao.org. Retrieved December 2, 2021, from https://www.fao.org/3/w1358e/w1358e14.htm
3. Agriculture|Coimbatore District, Government of Tamil Nadu|India. (n.d.). Retrieved December 3, 2021, from https://coimbatore.nic.in/agriculture/
4. Prabha D, Arunachalam R (2017) Constraints in adoption of mobile agro advisories by the farmers. Agriculture Update, 12(Special-7), pp 1782–1785. https://doi.org/10.15740/has/au/12.techsear(7)2017/1782-1785
5. Singh S, Ahlawatat S, Sanwal S (2017) Role of ICT in agriculture: policy implications. Oriental J Comput Sci Technol 10(3):691–697. https://doi.org/10.13005/ojcst/10.03.20
6. Swaminathan M, Swaminathan MS (2018) ICT and agriculture. CSI Trans ICT 6(3–4):227–229. https://doi.org/10.1007/s40012-018-0209-9
7. Ali M, Mubeen M, Hussain N, Wajid A, Farid HU, Awais M, Hussain S, Akram W, Amin A, Akram R, Imran M, Ali A, Nasim W (2019) Role of ICT in crop management. In: Agronomic crops, pp 637–652. https://doi.org/10.1007/978-981-32-9783-8_28
8. Bilali EH, Bottalico F, Palmisano OG, Capone R (2020) Information and communication technologies for smart and sustainable agriculture. In: 30th scientific-experts conference of agriculture and food industry, pp 321–334. https://doi.org/10.1007/978-3-030-40049-1_41
9. El Bilali H, Allahyari MS (2018) Transition towards sustainability in agriculture and food systems: role of information and communication technologies. Inf Process Agric 5(4):456–464. https://doi.org/10.1016/j.inpa.2018.06.006
10. Balkrishna A, Sharma J, Sharma H, Mishra S, Singh S, Verma S, Arya V (2020) Agricultural mobile apps used in India: current status and gap analysis. Agric Sci Digest—A Res J, Of. https://doi.org/10.18805/ag.d-5140
11. Anbarasan P (2021) A case study analysis on E-agriculture (e-velanmai): An ICT based technology transfer model in agriculture in Tamil Nadu state, India. Int J Current Microbiol Appl Sci 10(01): 1688–1696. https://doi.org/10.20546/ijcmas.2021.1001.197
12. Anbarasan P (2020) Psychological perspectives of farmers on mobile based agriculture: reuters market light (RML). The Pharma Innov 9(12S):136–139. https://doi.org/10.22271/tpi.2020.v9.i12sc.5467
13. Mathuabirami V, Makokha J, Karthikeyan C (2019) Constraints faced by extension officers of Coimbatore district, Tamil Nadu in using Uzhavan app. Int J Farm Sci 9(1):126. https://doi.org/10.5958/2250-0499.2019.00030.2
14. Richardson D (1996) The internet and the rural development recommendation for strategy and the activity
15. Shaik S, Helmers GA, Atwood JA (2005) The Evolution of farm programs and their contribution to agricultural land values. Am J Agr Econ 87(5):1190–1197. https://doi.org/10.1111/j.1467-8276.2005.00806.x

Secured Fog-Based System for Smart Healthcare Application

R. Hanumantharaju, B. J. Sowmya, Angel Paul, Ananya Muralidhar, R. Aishwarya, B. N. Shriya, and K. N. Shreenath

Abstract The increasing use of IoT devices and sensors can improve the effectiveness and efficiency of the healthcare system. These devices generally make very enticing claims. However, they should be utilized with caution as patient privacy, consistency, and efficiency are still real problems because cloud computing was not meant to handle the increasing volume, diversity, and speed of health data provided by IoT systems. It was deemed a preferred alternative to implement small data processing centers called Fog nodes, which would exchange data rather than send it to the cloud. Fog computing is an architectural technique that provides application-specific logic for network components between devices and the cloud. Latency, time, bandwidth, and many advantages may be alleviated, including declining latency,

B. J. Sowmya
Associate Professor, M S Ramaiah Institute of Technology, Bangalore, Affiliated to Visvesvaraya Technological University, Belagavi, India
e-mail: sowmyabj@msrit.edu

K. N. Shreenath
Associate Professor, Siddaganga Institute of Technology, Tumakuru, Affiliated to Visvesvaraya Technological University, Belagavi, India
e-mail: shreenathk_n@sit.ac.in

R. Hanumantharaju (✉)
Research Scholar, Siddaganga Institute of Technology, Tumakuru, Affiliated to Visvesvaraya Technological University, Belagavi, India
e-mail: rajurjs@gmail.com

A. Paul · A. Muralidhar · R. Aishwarya · B. N. Shriya
M S Ramaiah Institute of Technology, Bangalore, Affiliated to Visvesvaraya Technological University, Belagavi, India
e-mail: angelpaulm17@gmail.com

A. Muralidhar
e-mail: ananyamraju1234@gmail.com

R. Aishwarya
e-mail: aishuuu333@gmail.com

B. N. Shriya
e-mail: bnshriya@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_12

lowered bandwidth, heterogeneity, interoperability, scalability, security and privacy, real-time processing, and activity. In this paper, we make use of three-tiered architecture: the user tier, fog tier, and cloud tier. The user is authenticated and then is given access to enter the details, which are secured using advanced encryption standard (AES) algorithm. The fog computing layer's main functionality is collecting the data entered, securing, storing, and preprocessing them. This preprocessed data is sent to the cloud tier. The cloud tier performs predictive analytics using a random forest classifier algorithm. As a result, Fog may be considered a middleman between the cloud and the devices.

Keywords Fog computing · Predictive analysis · AES algorithm · Fog nodes · Health care · Cloud · End users

1 Introduction

What happens if the cloud isn't sufficient? This question arises a modern problem. According to experts, this decade will witness the beginning of a new technology known as fog computing. Fogging is the process of bringing cloud computing to the network's edge. It improves the collaboration between end devices and data centers. Fog computing is one solution to a number of problems. The global fog computing industry is expected to reach USD 753.67 million by 2025, according to various studies. Bringing cloud data closer to end devices eliminates delays in time-sensitive applications by ensuring that they are processed in a reasonable timeframe. Since all of the data does not need to be transmitted to the cloud for processing, we may use the fog layer to save bandwidth and money. Fog computing is important because it brings temporary storage closer to end devices with limited storage. Cloud computing is a temporary storage device, whereas fog computing is a persistent storage device. Over the past decades, the convergence among technology and health care has gained increasing attention all over the world. In the context of the health environment, safeguarding massive amounts of healthcare data collected via millions of objects, as well as maintaining the privacy and preventing unauthorized disclosures of such data during transmission across several sectors, is a major issue. Furthermore, wide, centralized, and universal access to medical data for authorized users at all stages of treatment or decision-making is a major issue. Some of these challenges are regarded to be beyond the capabilities of traditional cloud computing architectures. Furthermore, moving large volumes of disparate data to the cloud for storage or processing is impractical. Second, health-related decision-making processes need real-time interaction, mobility network support, and location awareness, all of which cloud computing lacks due to its inherent latency. Hence, feasible solution to address the above problem is seen in fog computing. A suitable approach for meeting the need is to add an additional layer between end devices and a distant cloud server. By preprocessing data, the additional layer known as the fog layer helps to reduce the volume of sent data for ensuring QoS and conserving network capacity. Thus, fog processing (Fog) has evolved as a viable option to accessing the Cloud, by moving computing and storage resources in proximity to applications and data sources and, as a result, avoiding delays. Healthcare applications are perhaps the most essential domain since they directly affect the lives of patients. Cloud-based solutions might cause delays in healthcare-related applications, which can lead to medical system failure and misdiagnosis. One of the main technologies that may be used in healthcare applications is fog computing.

In this model, we aim to achieve the collection and analysis of healthcare data to predict the user's current mental state and securing the obtained data using cryptographic algorithms (AES Algorithm). Since the existing systems have a lot of disadvantages, we are designing a system which secures the user's personal health records and also facilitates efficient storage management. The AES cipher is a symmetric key encryption algorithm which implies that a common key is used for both encryption and decryption of data. Data is converted into a format that cannot be understood or examined by anybody who does not have access to the key. Symmetric cryptosystems allow encrypted data to be transferred through a network even if it is likely that the data will be intercepted. But data decryption is impossible since no key is provided with the files. They need far less processing resources. As a result, symmetric encryption encrypts and decrypts data significantly quicker than asymmetric encryption.

2 Related Works

Though cloud computing is so prominent in today's world, emerging application domains such as IoT, smart phones, and automated devices attempting to use the features of eventual 5G networks necessitate a cloud extension to the edge, resulting in the new fog computing technology. Software systems work on both end devices and cloud and also on the intermediate nodes in Fog Computing. Fog computing as a new approach is a largely untapped field with so many unresolved research studies [1]. The combination of IoT and fog computing is expected to generate a lot of interest in the healthcare industry. As a result, patient care, customer satisfaction, and productivity improvement improved, with a favorable impact on all dimensions of health care [2]. Silva et al. [3] depicts a software framework based on fog computing that is intended to make health record management more efficient. It makes use of blockchain ideas to provide the crucial secrecy and to enable Fog nodes to perform the authorization procedure in a decentralized way. While cloud computing improves efficiency, it also has pitfalls such as data breaches and multiple cyberattacks. This allows an attacker to misuse data and use extremely sensitive data for illicit purposes. Thus, there will be a trade-off between the degree of these mechanisms and Fog node's performance. As a result, a balance between privacy and security and Fog layer's performance is necessary [4].

Predictive modeling is considered to be the most promising technique in the domain of e-mental health, which is a rich multidisciplinary area that provides novel research techniques. ANN was presented by Kellmeyer et al. [5] as a technique to protect large brain data from consumer-directed and clinical neuro technological devices. However, in order to produce reliable results, this model must be trained on a large amount of data. Yang et al. [6] developed an Internet of Things-enabled wearable gadget for psychological health care and an external equipment to capture voice data. This gadget would be able to detect pressure, motion, and a person's physiological state. With the introduction of machine learning and big data models, Passos et al. [7] believes that the long-established relationship between patient and doctor will vary. An impacted individual can use a machine learning system to track his fitness over time and alert his doctor if his symptoms worsen. An early check-up with a doctor might help the patient avoid a larger loss. If psychiatric sickness is not diagnosed or treated early, it compels the patient to engage in harmful acts such as suicide, as the majority of suicidal behaviors are linked to mental illness. Using a machine learning technique, [8] suggested a meta-analysis that targeted on suicide rate within one year after self-harm. Drug addiction is another adverse consequence of mental illness. By evaluating user data, early drug prediction can be made. Hasan et al. [9] investigated how Naive users acquire opioid use disorder using the MA APCD dataset. MTF data and Google trends were found to give joint assistance for identifying drug usage. Bardhan et al. [10] examined re-hospitalization of patients with congestive heart failure and built a model to predict when and how frequently they will be re-hospitalized.

Fog computing, according to Mahmud et al. [11], plays a vital role in supplying largely scattered end devices/sensors. As a consequence, fog computing has emerged as one of the most significant facets of research in recent years, from both an educational and an economic viewpoint. Ivan et al. [12] explores the current state of the art and discusses several broad challenges in fog computing such as protection, transparency, and integrity and process mobility across fog systems as well as between cloud and fog. According to Kumari et al. [13], fog computing facilitates service delivery without any delay with reliability while mitigating cloud computing challenges such as budget overheads, latencies, or disturbances while data is transmitted to the cloud. Furthermore, it is a decentralized horizontal topology that enhances the cloud's storage, computing, and networking capabilities. Security concerns in smart grids rely with concealing details, such as which equipment was used when as well as providing accurate brief information for appropriate invoicing. Lu et al. [14] structures multi-dimensional content using a super expansive pattern and encodes the structured data using the homomorphic cryptogram approach. Sarkar et al. [15] states that the outcome of a finding shows that in a situation where 25% of Internet of Things applications require services with minimal real-time latency, fog computing consumes significantly 40.48% less energy than the traditional cloud computing architecture. Existing systems must be improved in terms of scalability, network traffic, data management, service latency, and energy usage.

3 Methodology

Several fog architectures have been proposed in numerous studies. One of them is three-tiered architecture, which is currently the most popular architecture. As a result, fog may be considered a middleman between the cloud and the devices. The implementation involves configuring user tier, fog tier, and the cloud tier.

The following tools and technologies were used:

- Visual Studio: Used for the development of the entire system.
- Docker: To build and containerize different parts of the application.
- Oracle VM virtual box: Ubuntu Virtual Machine installed to perform data preprocessing and securing data using AES.
- Amazon ec2 instance: An ubuntu instance installed to perform predictive analysis.
- Flask: A framework to connect between the different layers.

Security and storage: Due of the rise of targeted attacks and persistent threats on medical data, the invasion of patient's privacy is becoming a serious concern. Medical records privacy is thus an essential aspect that must be taken into account. To ensure this in the user tier, the user is authenticated and then is given access to enter the details or view them. Requests that have been registered and authenticated are then mapped to the next tier. Outlier detection and private information data mining provide data security and privacy. Using standardized protocols and border protection methods, the communication protection module encrypts data transported between different layers with the help of the information flow protection module. The fog computing layer's main functionality is collecting the data entered, securing, storing as well as preprocessing them. This layer is used to extend cloud services to the device layer. Fog computing nodes provide the benefit of maximizing resource use. The fog data center helps to accomplish multi-tenant virtualization, enhance storage, compute, and other resource sharing demands in order to meet user expectations. Fog computing facilitates the storage and analysis of time-sensitive data on a local level. As a consequence, the distance and volume of data transported to the cloud are decreased, as are the security and privacy concerns associated with medical data. As a result, security and privacy concerns may be dealt with more promptly and locally. Fog nodes process data from users on a continuous basis. To send the preprocessed data to the cloud layer, these Fog nodes use a group of virtual computers.

In the fog layer, patient is presented with a questionnaire, from which the responses are collected. These are secured using advanced encryption standard (AES) which is symmetric algorithm that performs encryption and decryption. The AES encryption method outlines a series of reforms to be applied to the data. The initial step is to convert the data into an array, following which the encrypted modifications are performed over and over again in rounds. The algorithm is made up of four stages for each round. Substituting data in the array with is corresponding values from the substitution table by grouping bits is the first transformation, the second transformation is permutation step where all rows except the first are shifted by one in this stage, and the columns are mixed in the third transformation. The data is then XORed

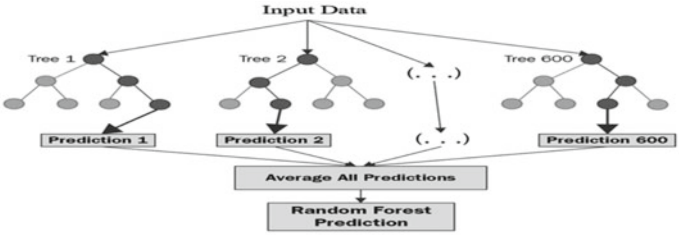


Fig. 1 Random forest classifier

with the appropriate round key in the last transformation. Longer keys require more rounds. The entered data is converted into cipher text with the key. The collected data and the key are stored in the fog layer and will be used for verification. Conversion to cipher text prevents illegal access to data as it isn't readable by human or computer. Further if the patient wants to view his details, a request is sent from the user tier. The patient is asked to enter the key that was used to encrypt his details. If the password matches with that of his previously entered key, the data is decrypted and displayed on the screen. According to security experts, AES is immune to brute-force hacks, which involves examining all potential key combinations until the correct one is identified. Encryption keys, on the other hand, must be large enough to prevent advanced computers from cracking them as shown in Fig. 1.

Predictive analytics: A person's state of mind has a lot of impact on his daily activities and his performance in his workplace. Depression, anxiety, and personality disorders are a few of the conditions that have a significant impact on the patient and are often ignored. The user tier of this system also has the service to anticipate if patient requires therapy or not based on his responses to the questionnaire. When a request is sent from the user tier to the fog layer to predict the patient's current state, the collected data is preprocessed. The data must be preprocessed before applying any of the predictive modeling approaches. This preprocessing step's main goal is to generate meaningful variables that can later be utilized in predictive modeling. This is carried out in the fog layer to reduce the load on the cloud tier. This preprocessed data is sent to the cloud tier. Predictive modeling aims to discover the model that delivers the accurate and reliable predictions, whereas statistical techniques focus on interpreting data in terms of correlation or identifying relationships. Predictions are based on what are known as characteristics or features, which are generated from observations. The cloud tier performs predictive analytics using random forest classifier algorithm. The random forest model is a predictive analytics ensemble model that uses an ensemble of decision trees to construct its model whose working is clearly depicted in Fig. 1. The objective is to pick the best and strongest model by polling a random sample of weak learners. The algorithm includes four steps, first step involves selecting random samples from a specified dataset. A decision tree for each same is constructed in the second step. The prediction responses are collected from each of these decision trees. Third step is to perform voting for every predicted outcome. Finally, the prediction result with the most votes is chosen as the final prediction result. While splitting a node, it searches for best feature rather than the most important feature among the randomly selected subset of features. As a result, there is a lot of diversity, which leads to an improved model. Advantage of this algorithm is that, on prediction it is easy to determine the importance of each feature. Random forest classifier enhances decision tree accuracy by reducing overfitting and tackles both classification and regression problems with ease. It can handle both categorized and continuous data and as it employs a rule-based approach, no data normalization is necessary. Random forest classifier automates the process of filling in missing values in data. While growing the trees, the random forest introduces more randomness to the model. When splitting a node, it looks for the best feature from a random subset of features rather than the most essential feature. As a result, there is a lot of variety, which leads to a better prototype. Random forest classifier has certain limitations. It necessitates a significant amount of computational power along with resources, because it generates several trees and merges their outcomes, and also it takes a long time to train because it uses a numerous decision tree to identify the class. It also lacks interpretability due to the ensemble of decision trees and is unable to assess the importance of each variable. The model is first trained with Mental Health in Tech Survey (Survey on Mental Health in the Tech Workplace in 2014-https://www.kaggle.com/osmi/mental-health-in-tech-survey) which describes mental health attitudes and the prevalence of mental health issues in the tech industry. It consists of 27 columns each of which is a query to know the patient's current mental state. There are 1259 tuples that are responses to these queries. The input features considered and its description:

Age-User's age

Gender-User's gender

Family history—Does the user have family history of mental illness

Work interference—Do you believe your mental health issue affects your ability to work?

Benefits—Is your company offering mental health benefits?

Care options—Are you aware of your employer's mental health care options?

Anonymity—Is your privacy protected if you choose to seek mental health or substance abuse treatment?

Leave—How simple is it for you to seek medical leave for a mental health problem?

To preprocess the training and testing data, columns other than the ones indicated above are dropped. Empty rows are filled with 0 or Nan for integer and string values respectively. For rows with age below 18 or above 120 are filled with median values. Different gender inputs are mapped to the corresponding gender, for instance if the user's input is m, it is replaced with male.

To train the model, the dataset is split into testing and training sets. Next step is to fit the random forest algorithm to the training set and each time a tuple obtained from the patient, it is appended to the test dataset and output is predicted. A value of zero suggests that the patient does not need therapy, while a value of one indicates that he should consult a psychologist.

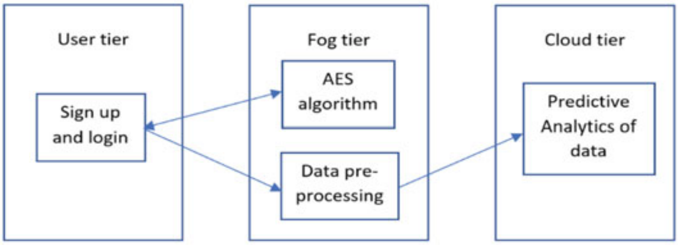


Fig. 2 Architecture of the model

The architecture of the model is shown in Fig. 2, for demonstration of the above architecture, the required components are a local machine with a virtual machine installed and an Amazon EC2 instance (virtual server to run applications). Here the virtual machine serves as the fog layer and the EC2 instance as the cloud layer. Each of these levels is communicate via requests from flask servers in each tier. Flask can easily be used to construct multiple apps and servers, each with its own functions which are distributed across multiple layers. This results in increased efficiency, performance, and testability. It also serves as a simple gateway to communicate between different layers. The user can interact with the system via a web application active on the local PC. Once the user has been verified, he is given access to enter, view his data, and perform prediction. The fog layer has 2 Docker containers running. Requests to enter and view the details are sent to the first container. This container performs AES encryption and decryption and thus helps in retrieving the details entered. The second container is invoked when the patient wants to predict if he is suffering from any mental health issues. This container has a flask application that performs data preprocessing and sends it to EC2 instance in the cloud layer. This instance contains a Docker container that has a flask server that listens to requests from virtual machine's flask server and performs predictive analysis on the patient's data. The output is displayed on the patient's screen.

4 Results and Discussions

The implementation of the model was successfully completed, and it yields fairly accurate results. The model consisted of the three-tier architecture which includes the user tier, fog tier, and the cloud tier. The sign-up and login activities were performed at the user end, securing the above confidential credentials using AES algorithm and data preprocessing were performed in the fog tier and the predictive analysis of the data was performed in the cloud tier. Initially, any patient who wants to use the Web site must sign-up and register through the sign-up page and later on each time the patient wants to use the Web site, he must authorize by logging in with his credentials. On successful authorization, the patient can either register by filling a form which has a certain set of questions which helps the doctor to analyze his symptoms and decides whether he/she needs to be treated or not, or the patient can view his medical records. The doctor can login using his credentials to either view the patient's medical history or analyze whether the patient needs treatment or not. The patient's data will be encrypted using AES technique and upon authorization the same will be decrypted and displayed to the end user.

To determine whether or not the patient requires treatment, we perform predictive analysis on the data using the random forest classifier technique. Figure 3 represents the normalized confusion matrix of the data that is taken from the patient. This matrix is a table which describes the classification model based on performance on a test data for which true results are often known, and it is easier to understand. The attributes used in our experiment are as follows, treatment attribute is used to determine if the patient has taken any prior treatment or not for the illness, work interfere is used to determine if his work adds stress and affects his mental health, family history is used to check if illness has any relation to his family genes, care options to determine if the patient has facilities to be taken care of, benefits attribute to ensure if he has any insurance schemes, obs_consequence checks for the consequence of the disorder the patient is suffering from, anonymity attribute ensures the privacy of the patient, mental health interview and wellness program attributes checks if the patient attends any interviews or program for his well-being, and seek help attributes checks if the patient wants help with respect to his mental health. The output of this matrix is used to evaluate the performance in terms of quality of the classifier in terms of the test dataset, higher the diagonal values, higher is the accuracy of the predictive results, whereas the off-diagonal values indicate that those are incorrectly labeled.

The matrix in Fig. 4 shows the normalized confusion matrix of all the attributes present in the dataset.

In the Fig. 5, we have a confusion matrix that helps determines the correctness of the model where 0 indicates the person is healthy and 1 indicates the person requires treatment. In the matrix, 133 indicates true positives (tp), 58 false positives (fp), 13 false negatives (fn), and 174 true negatives (tn).

Accuracy calculation: (tp + tn)/(tp + tn + fp + fn) * 100

Precision calculation: (tp)/(tp + fp) * 100

Recall calculation: (tp)/(tp + fn) * 100

F1 score: (2*Precision*Recall)/(Precision + Recall) * 100

Figure 6 consists of a histogram of predicted probabilities. The calculated results are given in Table 1. The histogram clusters the data into bins and is one of the fastest ways to visualize about the dispersion of each attribute in the dataset, it usually provides frequency of the observations that belongs to each bin that is generated for visualization purpose. The shape of the bin determines that the distribution is skewed. Analyzing the histogram generated, it depicts the count of the observations for which the person has to be treated or not where 0.0 indicates person is healthy and 1.0 indicates person needs treatment.

The graph in Fig. 7 is a ROC curve that depicts a classification model's performance across all conceivable benchmarks. It uses the true positive (tpr) and false positive (fpr) rates to plot the graph wherein:

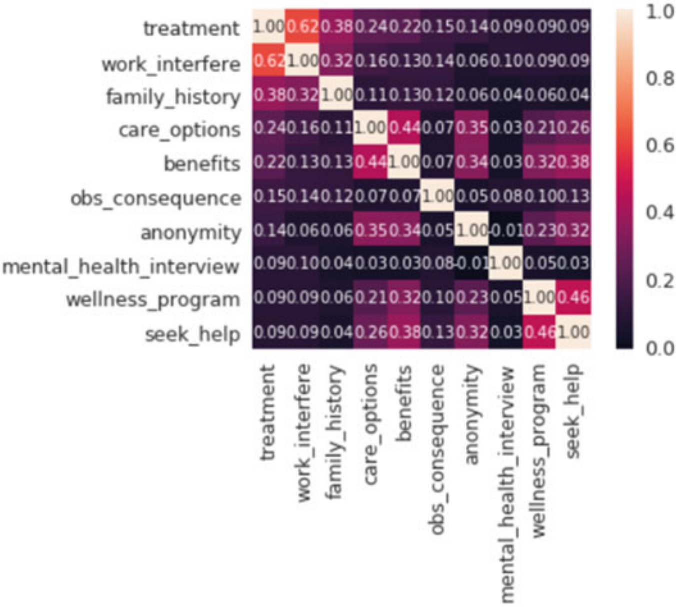


Fig. 3 Normalized confusion matrix for chosen attributes

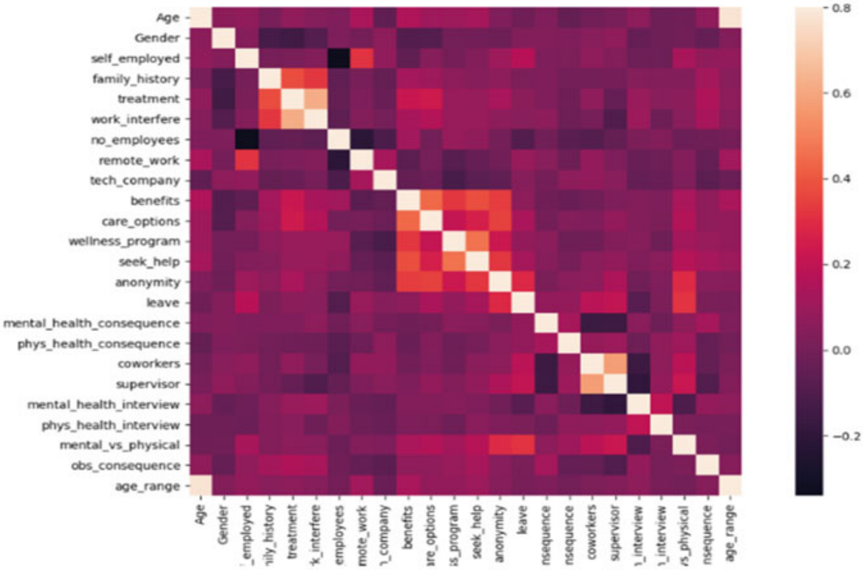


Fig. 4 Normalized confusion matrix for all attributes in the dataset

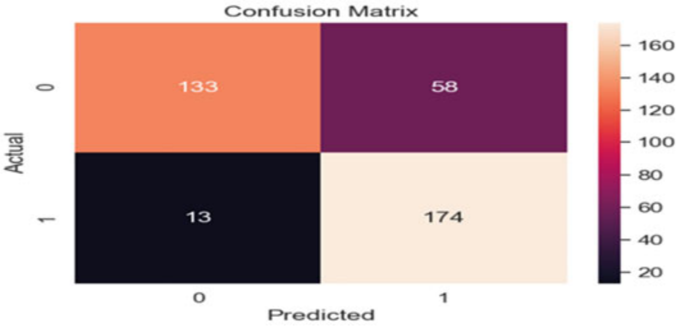


Fig. 5 Confusion matrix for calculations

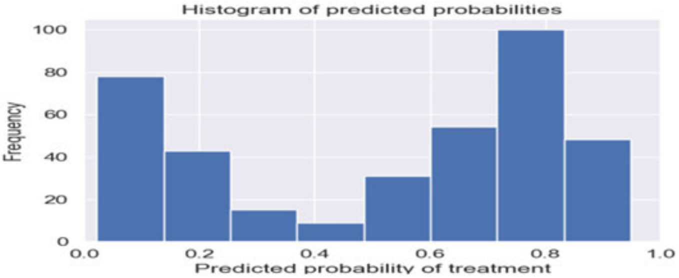


Fig. 6 Histogram of predicted probabilities

Table 1 Measured metrics

Accuracy (%)	Precision (%)	Recall (%)	F1 score (%)
81.2	69.63	91.1	78.93

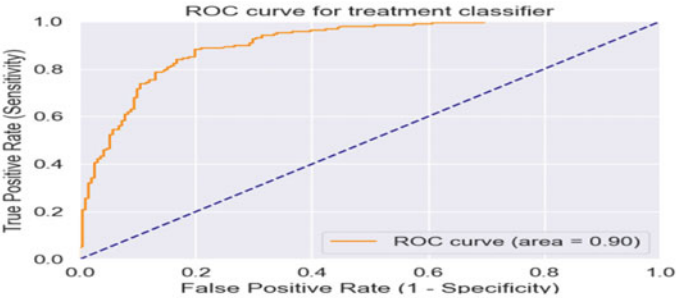


Fig. 7 ROC curve for treatment classifier

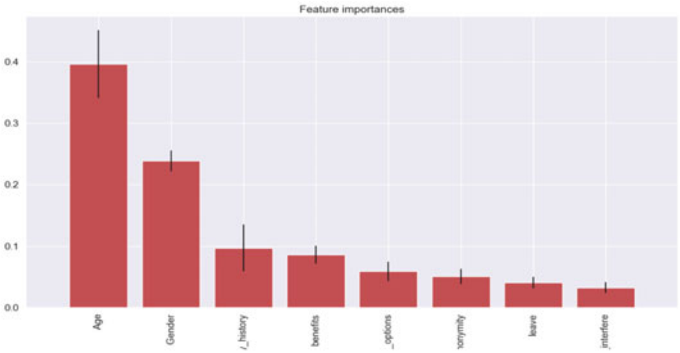


Fig. 8 Features importance graph

Tpr = tp/(tp + fn)

Fpr = fp/(fp + tn)

Area under ROC curve (AUC) is a metric that aggregates performance among all categorization criteria. AUC may be interpreted as the likelihood that the model rates a random positive case higher than a random negative example. A model with 100% incorrect predictions has an AUC of 0.0; one with 100% right predictions seems to have an AUC of 1.0. The model discussed in this paper has an AUC of 0.90 which means that the predictions are almost accurate. AUC is scale-independent. It assesses how well predictions are scored as opposed to their actual values. AUC is classification-threshold insensitive. It assesses the accuracy of the model's predictions regardless of the categorization level used.

The graph in Fig. 8 shows the weightage of the features considered for predictive analysis. The features present in the graph are basically the inputs taken from the patient which are as follows: age of the patient, gender, medical history, existing medical benefits, care options available, anonymity, leave taken, and work interfere. In the process of predictive analysis, the priorities of features differ where age is given utmost importance, and work interference is given least importance.

5 Conclusion and Future Scope

In the context of the health environment, the quandary lies in securing large healthcare data obtained from millions of objects, as well as preserving the confidentiality and prevention of unlawful disclosure of sensitive data during its transmission across various sectors. Maintaining a comprehensive, consolidated, and universal access to medical data for authorized users at all stages of treatment is a major challenge that is believed to be intractable by traditional cloud computing architectures, as moving massive volumes of varied data to the cloud for storage and processing is impractical. In this paper, we have implemented a three-tiered architecture consisting of the user, fog, and cloud layer with each layer having a flask layer. The findings of this study concluded that storing data in the fog layer reduces the access time and performing data preprocessing in the fog layer reduces the load on the cloud layer. Moreover, encrypting the data using the AES algorithm prevents illegal access of data from the fog layer thereby maintaining the confidentiality of private data. Fog computing avoids inefficiencies in the time-sensitive applications which come under the healthcare domain, wherein decisions need to be made promptly as it eliminates the need to establish contact with the cloud every time, decreasing the latency. As a result, the implementation of fog computing in this sector can prove to be extremely effective even as far as to help prevent medical system failures and misdiagnosis. Furthermore, the predictive analysis model could be extended to predict the mental illness the patient is suffering from and suggest a remedy or neighboring clinic should immediate attention be required.

References

1. Bermbach D, Pallas F, Pérez DG, Plebani P, Anderson M, Kat R, Tai S (2017) A research perspective on fog computing. In: International conference on service-oriented computing, pp 198–210. Springer, Cham
2. Khaloufi H, Abouelmehdi K, Beni-Hssane A (2020) Fog computing for smart healthcare data analytics: an urgent necessity. In: Proceedings of the 3rd international conference on networking, information systems & security, pp 1–5
3. Silva CA, Aquino GS, Melo SR, Egídio DJ (2019) A fog computing-based architecture for medical records management. Wirel Commun Mobile Comput
4. Alzoubi YI, Osmanaj VH, Jaradat A, Al-Ahmad A (2021) Fog computing security and privacy for the Internet of Thing applications: state-of-the-art. Secur Priv 4(2):e145
5. Kellmeyer P (2018) Big brain data: on the responsible use of brain data from clinical and consumer-directed neurotechnological devices. Neuroethics 11:1–16
6. Yang S, Gao B, Jiang L et al (2018) IoT structured long-term wearable social sensing for mental wellbeing. IEEE Internet Things J 6(2):3652–3662
7. Passos C, Ballester P, Pinto JV, Mwangi B, Kapczinski F (2019) Big data and machine learning meet the health sciences. In*:* Personalized psychiatry, pp 1–13, Springer, Cham, Switzerland
8. Kessler RC, Bernecker SL, Bossarte RM (2019) The role of big data analytics in predicting suicide. In: Personalized psychiatry, pp 77–98, Springer, Cham, Switzerland
9. Hasan MM, Noor-E-Alam M, Patel MR, Modestino AS, Young G, Sanchez LD (2019) A novel big data analytics framework to predict the risk of opioid use disorder
10. Bardhan I, Oh JH, Zheng Z, Kirksey K (2015) Predictive analytics for readmission of patients with congestive heart failure. Inf Syst Res 26(1):19–39
11. Mahmud R, Kotagiri R, Buyya R (2018) Fog computing: a taxonomy, survey and future directions. In: Internet of everything, pp 103–130. Springer, Singapore
12. Stojmenovic I, Wen S (2014) The fog computing paradigm: scenarios and security issues. In: 2014 federated conference on computer science and information systems, pp 1–8. IEEE
13. Kumari A, Tanwar S, Tyagi S, Kumar N (2018) Fog computing for healthcare 4.0 environment: opportunities and challenges. Comput Electr Eng 72:1–13
14. Lu R, Liang X, Li X, Lin X, Shen X (2012) Eppa: an efficient and privacy-preserving aggregation scheme for secure smart grid communications. IEEE Trans Parallel Distributed Syst 23(9):1621–1631
15. Sarkar S, Misra S (2016) Theoretical modelling of fog computing: a green computing paradigm to support IoT applications. Iet Netw 5(2):23–29

Meta-Heuristic Optimal Path Planning in Blockchain-Aided UAV Swarm Network

M. Kayalvizhi and S. Ramamoorthy

Abstract Unmanned aerial vehicles (UAVs) are rapidly developing in major fields but also exposed it to security vulnerabilities. UAVs operate in high airborne network where it encounters many obstacles and uncertainty. And the existing path planning algorithms don't consider path deviation attacks where the data communication is not secured. But, in multiple unmanned aerial vehicles, path planning is a distinctive optimization problem. And for swarm of unmanned aerial vehicles (UAVs) deployed in an uncertain dynamic environment, optimizing the UAV coverage area is a major research area that it needs to be addressed to support effective and secure data transmission. For optimized path planning in UAV swarm, we proposed a modified particle swarm optimization (MPSO) algorithm which encodes the multidimensional points for instant optimal path. The factors that are considered for the encoding helps to find the global optimum. And the blockchain distributed ledger in UAV nodes will protect the UAV network from security attacks such as denial-of-service and man-in-the-middle attacks. It also supports unrestricted coverage area, where the storage is made active only on demand which is dynamically cost-effective in UAV network. Simulation results shows that the proposed algorithm provides instant optimal path with unrestricted coverage aided with data security.

Keywords Swarm optimization · Denial-of-service attack · Man-in-the-middle attack · Blockchain · Data security · Unmanned aerial vehicles

1 Introduction

For operating in high resilient-based data communications in hostile environments, unmanned aerial vehicles (UAV) are qualified to provide the required service. The major monitoring environments include both ground-level monitoring and space-air-ground-level monitoring. And another major area includes monitoring in agriculture

M. Kayalvizhi (✉) · S. Ramamoorthy
SRM Institute of Science and Technology, Chennai, India
e-mail: km3833@srmist.edu.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_13

and thermal sensors-aided monitoring. And the UAV can operate in various communication channels in both 5G [1] and 6G. UAV can provide is high potential service in these communications area. And the hostile environments are monitored via UAV in larger area which should be aided with network security. To acquire the facts about the objects such as both static and dynamic, the unmanned aerial vehicles are in motion randomly across the environment. So, to regulate the operation between the terrestrial and the air medium, an unmanned aerial vehicle-enabled communication system will be a good option. But, the main gap lies in choosing of optimal path and choosing a dynamic optimal path where the UAVs should be capable of deciding optimal path under dynamic conditions, where the UAVs will not have complete information to perform the optimal path planning. So the result of optimal path planning is to provide a better shortest path, where the UAVs can deliver the collected information to the destined data sink or base station. However, the delivering of data to the data sink will not be smooth, since the UAVs operating in dynamic environment under uncertain conditions will face signal interruption and selection of the path. And the monitoring of relay networks will be an additional overhead and in turn affect the data quality and delivery.

The successful data communication in unmanned aerial vehicles is based out on the relay state of data and the routing path. And during path up-gradation, radio dissemination can be used to broadcast the data. And when the UAV communication system is exposed to security attacks such as jamming attacks, it has the capability to remove the ground control station from the UAV system, and the attacker can gain control to the base station and compromise the network. So, the UAVs that are participating in the network needs a secure mechanism to transfer data among themselves and to the base station since the data can be eavesdropped by an attacker which leads to compromising the mission. And when it comes to denial-of-service attack, most attackers will try to drain the UAV's energy and power, which can compromise the mission. To make the UAV network resilient from network attacks, each UAVs in the network needs to be authenticated, and only, authenticated UAV nodes are allowed to communicate in the UAV network. And the existing algorithms for UAV path planning are feasible only when the UAV's power resource is at an optimal level and once the power resource is drained the path planning cannot be completed. The path selection algorithm should be in controlled phase for efficient optimal path selection. And in the most traditional wireless network for optimal path selection, OSPF protocol is incorporated. And the existing OSPF methods don't satisfy successful data delivery, data quality, and time. And the trajectory planning didn't protect the network from jammer attacks. The UAV motion such as upgrading angle and projection was not restricted in traditional methods, but however, the UAVs performance was not improved in the traditional path planning methods which are firefly algorithms, particle swarm optimization, and swarm heuristics algorithms. The upgraded routing path should provide the angle of climb for communicating in relaying state. So, in order to mitigate these issues, we are proposing multipath selection with secure data communication which will improve the UAV system performance. And the path selection in the UAV system which can support under dynamic uncertain conditions is considered as a novel approach to remove the gaps in the traditional methods. The algorithm proposed here is meta-heuristic multipath algorithm which mitigates the problem of both path planning and path update. And the simulation results show that the proposed model will enhance the UAV system data transfer under high dynamic uncertain environment.

When highlighting the uncertain environment, the UAV energy and the performance can be affected by external environment factors. These uncertain factors are more in high mobile dynamic airborne network. The main uncertain factors are a heavy gust of wind which makes the UAV navigation difficult, air density, ambient temperature, gravity, and the surveillance areas such as terrain and caves which have unpredictable hindrances and obstacles which in turn affect the performance of the UAV path planning.

The rapid growth of technologies in wireless network also led way to too many security attacks in terms of securing data. When considering data security in UAV swarm network, the most promising solution will be blockchain technology. So, generally, in blockchain, the data are broken as blocks where each data block consists of data, time of block created, and its own hash. So, when the data are chained as blocks interconnected with the hashes, it is not possible to alter the data in a block. And even if an attacker tries to change it, the hash value gets changed, and the rest of the blocks will become invalid. So, for successful compromise of the network, the attacker needs to takeover 50% of the network which is possible only if the attacker has huge computational power and resource.

The paper is written as follows. Section 2 focuses on existing works; Sect. 3 elaborated on the proposed meta-heuristic path planning algorithm. Section 4 provides the constraints in path planning. Section 5 focuses on the security attacks in UAV swarm. Section 6 presents the simulation results and validates the performance of blockchain. Section 7 discusses on the simulation results. Section 8 provides the conclusion.

2 Existing Works

In this part, brief details on the existing path planning algorithms in UAV and network security are discussed. The various existing algorithms that support getting optimal path in the UAV network.

Patle et al. [2] The author uses firefly meta-heuristic algorithm for optimal path planning. For navigating of automatic robot issues, the firefly flashing method is applied to follow the track the path when the source is static and dynamic. Firefly algorithm needs tuning of parameters to operate effectively. It shows reduction in speed since the firefly algorithm will be efficient only in smaller distance since the algorithm works only based on light and attraction which gets absorbed. This proposed algorithm is not very suitable for handling complex problems.

Gjanci et al. [3] The author provides a solution to identify the path of the UAV which will deliver the data to the sink which it can be maximized. A greedy and adaptive AUV path-finding (GAAP) heuristic is defined that guides the autonomous vehicle for data collection from nodes. The energy efficiency and communication is achieved only by reducing the link length. UAVs need to be operated at a range of over 200 km where this proposed algorithm will not be feasible.

Sorbelli et al. [4] The author proposed a method to handle spoofing attacks where by using vision chips in single and multiple UAVs. When focused on GNSS+, no attacks were studied, and a solution was proposed. For preventing path deviation attacks, secure communication model needs to be employed in UAV and base station communication.

Chen et al. [5] The author focuses on denial-of-service attacks that target UAV network by continuously monitoring the UAV system and its activities. But, this will cause an additional overhead, and the system resources will be drained. So, if the resources are drained, then the UAV operations cannot be handled. To handle the resource constraints, a secure cloud storage system needs to be deployed.

Xu et al. [6] The author targets on UAV assisted by IoT where it is adopted in all major areas. But here, since the network speed is increased, it degrades the data communication. It also incurs privacy issues. So, the proposed work will not be suitable in uncertain dynamic environment.

Wu et al. [7] The author focused on an iterative algorithm which enhances the consumption of energy and throughput. The proposed work gives optimal path which solves the non-convex problem. But, this algorithm will not be able to give optimum results in uncertain environment.

Wang et al. [8] The author has proposed algorithms to meet the requirements of convergence rate, accurate and cross-over searching. The main limitation of PSO algorithms is it is less practical and not feasible when applied to complex scenarios. The proposed solution will not cover high mobility access points since it prematurely gets traps into a local minimum in uncertain environments.

Liu et al. [9] The author developed a selection policy-based algorithm which is constructed based on deep q-network. The dynamic framework which is proposed will be suitable only for single UAV and doesn't focus on high mobility environment. The proposed solution doesn't focus on multi-UAV network.

Guoliang et al. [10] The author presented ant colony algorithm which is an improvised version to mitigate the drawbacks on path planning. Ant colony algorithm probability distribution changes with iteration where the computation time is undefined. The proposed algorithm doesn't focus on uncertainties in high mobility dynamic environment.

Jia et al. [11] The author presented a dynamic programming framework to obtain optimal visiting order of the node and data collection strategy. The proposed dynamic framework will be suitable only for the fixed nodes where the location doesn't change over time. The proposed solution will not be feasible when used in high mobility dynamic environments and doesn't support incorporation of multi-UAVs.

Govind et al. [12] The author presented a first depth-first search (DFS) algorithm to select fly cell sequence where the UAV visit to collect data. Post-selecting sequence, NSGA-II meta-heuristic algorithm used for optimal path. The use of DFS algorithm needs to determine the depth until the search has proceeded. Its complexity increases when the node distance is high. And chances are there that it may go down the left-most path forever. The proposed solution of NSGA-II and DFS doesn't support swarm of UAVs.

Chen et al. [13] The author proposed an improved A* algorithm. One major practical drawback of A* algorithm is its space complexity, as it stores all generated nodes in memory. Here, the proposed improved A* algorithm strives to find optimal path by simplifying the search space, and it concentrates only on single UAV path planning. The improved A* algorithm solution doesn't support multi-UAV network.

Chawra and Gupta [14] The author proposed a Salp-swam optimization method based on meta-heuristic. The proposed Salp-swarm optimization often encounters low convergence speed and getting stuck in local optima, and the differential evolution algorithm result will not be optimal, and possibilities are there it will get path that is not optimal. The proposed solution applies only to cluster-based WSNs, and the cost of clustering is an important parameter to authenticate the effectiveness of the scheme.

Yoon et al. [15] The author proposed mTSP algorithm. mTSP-GA which is used to find path that is optimal. Author used a zigzag path via K-means. mTSP genetic algorithm consumes time, especially for scenarios with a large number of uncertain factors in high mobile environment. The proposed work will not anticipate cluster value, K–value, and the initial value which will influence the output which will not be feasible in dynamic environments.

Binol et al. [16] The author proposed modified evolutionary method. Though genetic algorithms are easy to implement, it can be time-consuming, especially for scenarios of high mobile dynamic environment. The main drawback of harmony search is it may get stuck into local solutions and has premature convergence and slow convergence speed.

Yanmaz et al. [17] The authors surveyed on deterministic and probabilistic path planning algorithms its limitations and advantages. The study results showed that the approach of deterministic consumes more deciding time and only generates results on probabilistic scenario.

The UAVs are capable of communicating in massive scenarios where there will multiple inputs and multiple outputs [18, 19]. And the UAV communication is supported by cellular networks. And the cellular network communication will also be facing multiple challenges for effective operations [20]. So, the optimal path planning is essential in massive communications. The existing works in the path planning of swarm UAV's such as iterative algorithm, firefly algorithm, GNSS, deterministic algorithm, probabilistic algorithms, MTSP focus in static network path selection. The existing algorithms cannot provide an optimal path in a dynamic environment. These algorithms cannot mitigate data loss and deviation of path attacks which can compromise the mission. The comparison of existing algorithms and their drawbacks is mentioned in Table 1. So, the research gap identified is to provide a secure environment for the UAV swarm with optimal path planning. So, in this paper, we are proposing meta-heuristic path up-gradation path planning model.

Table 1 Comparisons of existing algorithms

Algorithm	Behavior	Limitations	Performance in a dynamic environment
Firefly algorithm	Gathering	Parameters do not change with the time	Partial environment
Ant colony algorithm	Foraging and trailing	Slow convergence	Restricted area in dynamic environment
Bee colony algorithm	Foraging	Destination reached only when population size is increased	Restricted coverage area
Particle swarm optimization algorithm	Aggregating and flocking	premature convergence, trap in local minima	Restricted area in dynamic environment

Table 2 Parameters in UAV network simulation

Parameter	Description	Value
(X:Y)	Unknown values	(22.48; 2.35)
p	Cost of packet loss	10x 10^3
λ_{nu}	High position range	22~34 dB
α (h1)	Height of UAV	15m
δ(d1)	Depth of UAV	15m
$R_{\text{d}r}$	Data rate of UAV (25Mbyte/s)	170 Mbps
UAV_c	Radius of coverage in UAV	100~2500 m
Ep.no	Number of episodes	12
N	Number of UAV nodes	14
S	Number of UAV server	1

3 Path Planning with Meta-Heuristic Algorithm

Unmanned aerial vehicle with a heuristic system enhances the environment with uncertain factors in UAV network with high mobility. The features categorized into optimal path planning aided with secure transfer of data.

The unmanned aerial vehicle network is incorporated with a heuristic optimization system that can support uncertain environment under both static and dynamic UAV network. And the main proposed method can be categorized into heuristic optimized path planning and securing the data communication. The proposed meta-heuristic optimized path planning algorithm is developed to operate under dynamic UAV network which is supported to operate with the help of base station and doesn't fully depend on the base station which can navigate the UAV to the destined location. Here, range has no constraints. For evaluating the proposed algorithm, the following parameters are considered which is encoded at multidimensional points. The parameters are intra-UAV path, throughput, energy status of UAV, and loss of packet. The proposed algorithm provides exact distance between UAVs which can reduce the relay data transfer among the UAVs. The function of evaluation is derived as sum of fitness as follows:

$$\begin{aligned} &\mathrm{In} = \beta \times \mathrm{UAVn}/\sum \mathrm{i} = \mathrm{ie1} + \beta 1 \times 1/\sum \mathrm{i} = \mathrm{ie2} + \beta 2 \times 1/\mathrm{Tuav} \\ &\left(\sum \mathrm{i} = \mathrm{m}\left(\sum \mathrm{i} = \mathrm{iey1} + \sum \mathrm{i} = \mathrm{iey2}\right)/\mathrm{Tuav}\right) \end{aligned} \tag{1}$$

In Eq. (1), the distance between the UAV nodes is calculated, and also, the node energy level is tracked which is updated in the routing table. And after getting the destination details, the UAV nodes in the UAV network are updated with the details of distance, location, and the energy level of all the participating UAV nodes. The proposed optimization algorithm will be able to collect the data from the dynamic environment with uncertain factors. And the UAV network node is evaluated from Equation (2). This equation helps to find the data capacity of each UAV node which is used for optimal path selection.

$$\mathrm{PV} = \beta 1 \times \sum \mathrm{k} = 1 \cdot 1 = 1\left(\sum \mathrm{i} = \mathrm{ie1} \sum \mathrm{i} = \mathrm{ie2}\right) + \beta 1 \times \sum \mathrm{i} = 1\ \mathrm{B\ dc} \tag{2}$$

The path value (PV) provides UAV distance. And the UAV node sensing quality is evaluated from the collected data capacity which is evaluated using PV.

The UAV network evaluation method implemented with meta-heuristic optimized path planning where it can support multiple UAV nodes that are integrated under dynamic environment that ensures both data and UAV privacy and security. The evaluation method is responsible for calculating the PV. The research gaps in the existing algorithms such as localization problem and computational errors are overcome by the PV evaluation method. The proposed algorithm selecting values are used as deciding factors in UAV network provides optimal path. The proposed algorithm is mathematically optimized that is used to select the instant path with limited computation.

4 Path Planning Constraints

The evaluation factors in the path planning method aid the UAV nodes for updating the searching path. The searching path update tracks the following parameters such as UAV identity, location of the UAV, and the next hop duration. These parameters help to provide an updated path with secure connection between the UAVs. These values are updated in real time continuously. These values constitute updated path routing table. This evaluation methods help to address the issues such as path planning convergence, multipath distortion, and matrix decomposition issues.

In order to complete the UAV swarm operation, the UAV node that needs to reach the destination doesn't have energy to continue the data communication and communicate with the neighbor UAVs or perform data exchange. And the UAV network system should ensure that the UAVs energy level is not drained, and it maintains the threshold level. The routing table maintains the updated PV values where the network stability is achieved. And the trajectory planner applied on the decision node helps to provide the direction based on the index of the parameter. During the UAV communication, if the data transfer is not possible for direct transfer, then it can be done via indirect communication. And the routing table is updated with the current PV to construct the indirect path.

The UAV server is supported with blockchain technology [21] which takes responsibility to transfer the data. And the identities of all UAV nodes are maintained via the UAV server. And to improve the data communication, the UAV server is implemented with enhanced interior gateway routing (EIGRP) protocol since the higher bandwidth enhances the connectivity. The proposed blockchain model secures the data transfer in UAV environment. The EIGRP protocol implemented in the UAV server supports x86 architecture. And the data security is achieved by using asymmetric encryption.

The EIGRP protocol implemented in the UAV server keeps updating the routing table which provides the instant path where the UAVs will be hovering in their position till they receive the updated path details. Each UAV node has a distributed ledger which received the data that are encrypted. UAV nodes send the collected details to the UAV server which will be encrypted, and the UAV server checks the received data and shares the updated routing information. The proposed meta-heuristic optimized path planning algorithm will send data to the UAV server where server will decrypt the data. The ledger maintains the details of the activities done in the UAV swarm network. So, when a malicious UAV node breaks into the UAV swarm, it will not be aware of the updated routing table implemented in the UAV server. So, the malicious UAV node energy level will be depleted with its movements, and it will be rejected by the network since it doesn't have valid key.

5 Denial of Service Attack in UAV Swarm

Denial-of-service [22] attack in UAV network scenario is discussed. Uncertain UAV network is depicted in Fig. 1 which depicts a UAV network that is secured.

5.1 System Model

The proposed system is referred as client operating system where the security system is monitored. The proposed heuristic optimization algorithm is responsible for providing instant path selection aided with data security. The position of the UAV node is encoded under multidimensional points, so all the best ways are analyzed, and

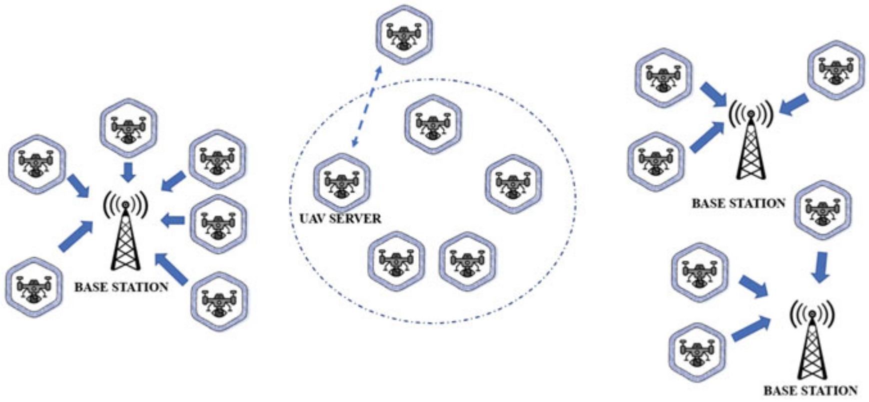


Fig. 1 UAV node managed by UAV server under uncertain environment

only, the best optimal path is added into the routing table. The proposed blockchain architecture completely supports the uncertain environment since the update routing path is shared to all the UAV nodes and saved in the distributed ledger which it can be mitigated from the denial-of-service attacks and promises a secure UAV network [23]. The malicious activities are monitored regularly which keeps the UAV swarm network secured [24].

5.2 Attacker Model

The UAV swarm network cannot support data security under uncertain environments. The malicious attackers try to embed themselves to the network, which tries to compromise the network. So, the client operating system suffers from security attacks in uncertain environments. So, the proposed system should be able to protect the network from denial-of-service attacks which can affect the performance of the network [25].

5.3 CPU Denial-of-Service Attack Protection

The CPU resource management should be utilized properly to provide optimized coverage in the UAV swarm network. The CPU utilization varies based on the real-time applications scenarios. So, the requests need to be prioritized based on the first come first serve. So, a high priority task is completed first where the low priority task is completed next [26]. So, by this way, the CPU utilization can be properly utilized.

5.4 Memory Consumption and DoS Protection

The abnormal movements of UAV in uncertain environment which is due to denial-of-service attacks can drain the memory resource in the UAVs which can affect the network performance. But, when the CPU utilization feature is deployed, the memory measurement issue cannot access the memory excessively. The role of optimization controller tracks the network performance monitors access of memory. So only, during this period, the memory is accessed. And the memory issues need to be resolved in the UAV network for effective performance [27].

5.5 Data Communication DoS Protection

The controller of optimization collects the sensor data and navigates the UAV nodes in the network for effective operation [28]. This helps to secure the system from DoS attacks. This helps to take control in the uncertain environment and monitors the network periodically. The denial-of-service attack is performed on the swarm of unmanned aerial vehicles where the malicious UAV node tries to authenticate itself as legitimate node and tries to join the UAV swarm network. But, in our proposed method, the UAVs are registered in blockchain network where each UAVs have a valid key to identify itself and also the routing table is maintained in order to monitor the traffic of the UAV nodes. So, when the attacker UAV node, which is not aware of the routing table and the registered key, tries to drain its resource, and it is denied by the UAV network. So, the proposed blockchain architecture is capable of identifying the denial-of-service attacks. Figure 2 shows how malicious UAV node is rejected by the UAV swarm network.

5.6 Simulation Mode

The sensors in the UAV network should be secured activities that are malicious from vulnerable applications. The optimization controller doesn't restrict the access of memory but will track the network details of the system. The network activities are transferred from the sensor nodes to the optimization controller. So, any change in the movement of the UAV's which is not in the routing table will be alerted to the optimization controller, and the malicious node can be rejected from the network. This can prevent it from DoS attacks.

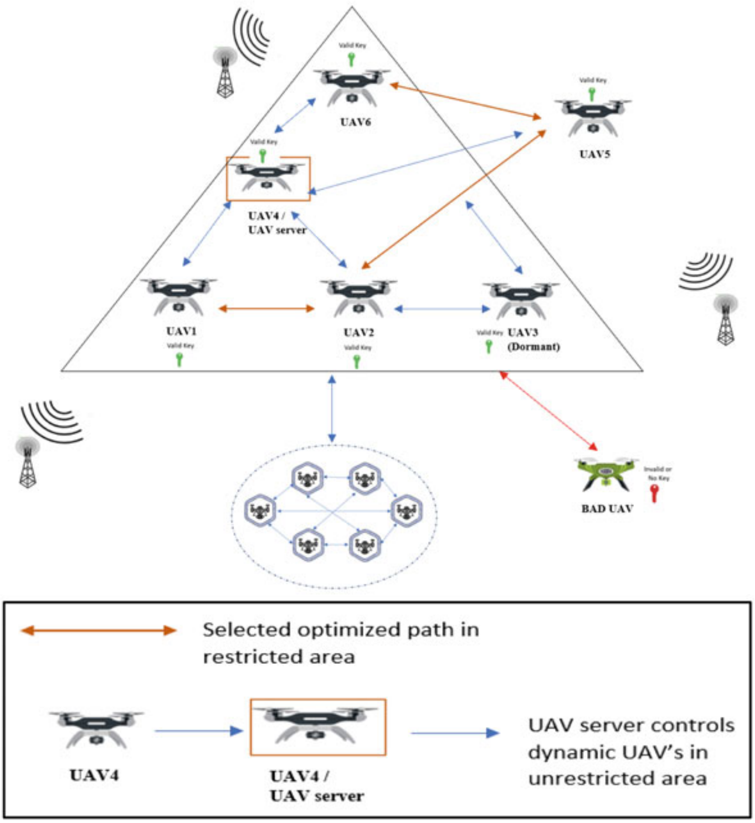


Fig. 2 Malicious UAVs rejected from UAV swarm

5.7 Data Security Monitoring

The blockchain network in an uncertain environment is illustrated in Fig.3. The network connectivity is maintained by the controller of the UAV server. The protect from malicious data transfer in the uncertain network can act as a base station and can invoke the meta-heuristic optimization algorithm for path planning. So, whenever a UAV node is moved to a space that is out of the initial coverage area, the optimization controller will acts as a base station, and the path details are updated in the routing table to support the UAV node that is out of coverage. So, the updated routing table helps to control the communication network for unrestricted coverage. Blockchain network verifies the authenticity of the user for accessing the data and controlling the data transmission. The proposed timestamp is validated to check if the user has done any activity in the UAV network at a particular time since the details of all the UAV nodes activity are tracked in the routing table details.

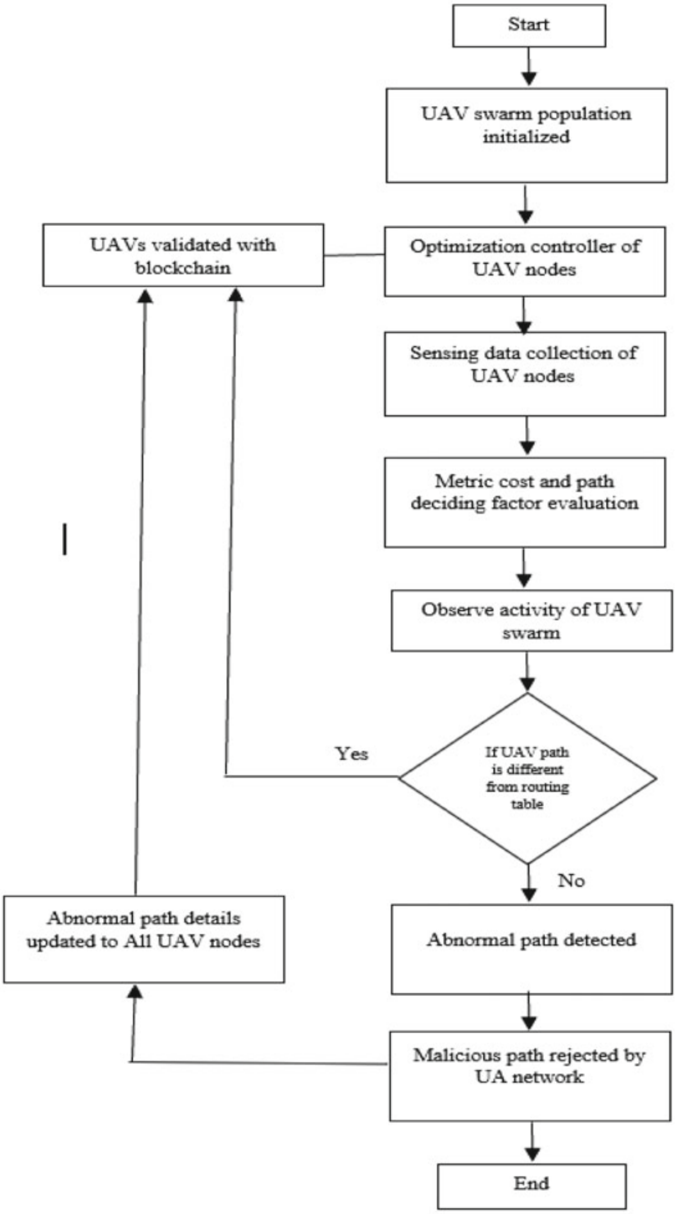

Start
UAV swarm population initialized
Optimization controller of UAV nodes
UAVs validated with blockchain
Sensing data collection of UAV nodes
Metric cost and path deciding factor evaluation
Observe activity of UAV swarm
If UAV path is different from routing table
Yes
No
Abnormal path detected
Abnormal path details updated to All UAV nodes
Malicious path rejected by UA network
End

Fig. 3 Uncertain environment scenario aided with blockchain network

The modified swarm optimization algorithm method supports dynamic UAV movements to identify the positions in dynamic search space. When the destination is identified, the sender confirms that it is not smaller than a pre-defined position. The optimization controller senses the UAV network continuously and identifies the threat activities. And the interval for the continuous monitoring is set to a normal threshold which is an average value. If the threshold exceeds, then the behavior is set as unusual, and it can be rejected. The threshold value should always be set to an average value to detect the abnormal behavior. The dynamic search space is identified provides the destination target where the proposed path planning algorithm covers uncertain environment with unrestricted coverage.

6 Experimental Setup

The experimental results supporting the proposed work are presented in this section. The proposed algorithm is compared with the existing GNSS algorithm, and the results are evaluated. The simulation setup was performed in ContikiCooja where the coverage radius is of 4 km radius. The data constraints used in this simulation experiment are listed in Table 2. The bandwidth for each node is 15 Mhz. And the UAV server is also assigned with 25 MHz bandwidth to avoid congestion. The range of spectrum is 2.8 where the UAV nodes are positioned geographically. Since the UAV nodes are operated in high dynamic environment with uncertain factors, it is defined as x= 23.35 and y=2.35 at 5 GHZ frequency. For each node, the packet cost is the associated uncertain factors. The value of UAV node network location is assigned based on the experimental simulation.

6.1 Path Selection Scheme Evaluation

In this section, the proposed algorithm is evaluated to get the optimal access path. The UAV server is executed. Here, all the UAV nodes are maintained by the UAV server, and all the UAV nodes are under the control of the UAV server. The UAV server accesses all the UAV nodes which acts as a multipath flashing scheme. The automatic optimal path model provides optimal path from all the existing paths. The existing heuristic algorithms are difficult to get the optimal path due to unrestricted coverage area. The coordination system such as straight-line transformation is used to reduce the UAV computational cost. The existing methods have high costs due to the methods which are complex. From Fig. 4, the radius coverage is fixed as 1000 m where the coverage radius is getting higher than the fixed value.

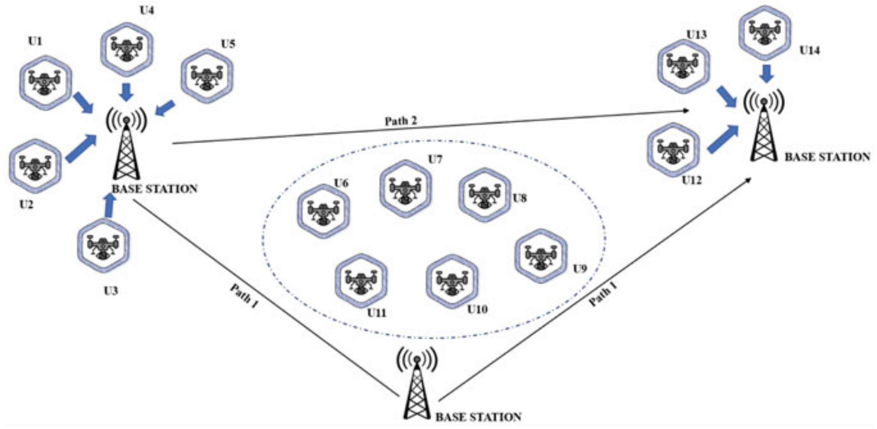


Fig. 4 Data communication from source to destination in UAV network

6.2 Multi-Path Flashing Model

The routing table maintains the activity of the routing path and keeps it updated. And the routing table details are shared only to the source, and the destination and the positions are arranged based on the values of the routing table. If there are fewer UAV nodes, chances for security attacks are high. But, if multiple UAV nodes are participating in the network, the proposed blockchain architecture doesn't allow to spoof the UAV nodes. So, the optimization method provides defense localization. From the results, the location of UAV node is secured, and the distance details are also secured [29]. The proposed blockchain-based architecture prevents the model from the spoofing attacks. The real-time UAV operations can be open to many security attacks. But, the proposed defense model will be able to monitor any abnormal behavior of the UAV nodes and will be able to block any kind of malicious activity [30].

6.3 Path Planning Algorithm Execution

Simulations were performed for the proposed algorithm with EIGRP and tested in the following scenarios. The test setup observes path planning of UAV nodes when data packets are sent from one UAV node to next UAV node. So, from the routing table which is updated, the UAV nodes follow the optimal path where the computation cost is low and data loss is low.

The proposed modified particle swarm optimization is able to provide instant path generation in uncertain conditions. The routing table is updated in real time to track the details of each UAV node. And the energy level and the neighbor distance are tracked to find the best optimal path. The routing table and the low delay value calculated support to choose the optimal path. And the blockchain-aided architecture will be able to support the identification of the UAV nodes. The experimental model was tested with EIGRP protocol, and the low time delay and low cost prove that the algorithm which is proposed is preferable than the existing algorithms, and the path is updated instantly from the execution time. The UAV nodes of U1 (UAV1) to U15 (UAV15) are deployed in the simulation model. The algorithm that is proposed is tested in the scenario where data packets sent from U1 to U11 and from U2 to U12 and from U3 to U13, and from U4 to U14 and from U5 to U15. These five scenarios are tested in the simulation mode, and the results are compared (Table 3).

Metric value is calculated for path planning which is used for data transmission from the source to the destination. Once the metric values are calculated, the routing table is updated with the best optimal path which is selected based on the metric value. This metric value is sent through a decentralized blockchain model [31, 32] for secure data transmission. The metric value evaluation and the updated routing table supports for instant path selection with normal computational delay and cost when compared to the existing algorithms. The proposed algorithm shows better results in terms of unrestricted area coverage and automatic path planning.

Table 3 Delay value comparison

Delay value comparison of path up-gradation (seconds)						Total value
Methodology	Scenarios					seconds
	1	2	3	4	5	Delay value
EIGRP	0.00624	0.00533	0.00425	0.00322	0.00214	0.004236
Existing Routing	0.01235	0.01124	0.01123	0.01120	0.01024	0.011252

Table 4 Meta-heuristic path planning result comparison

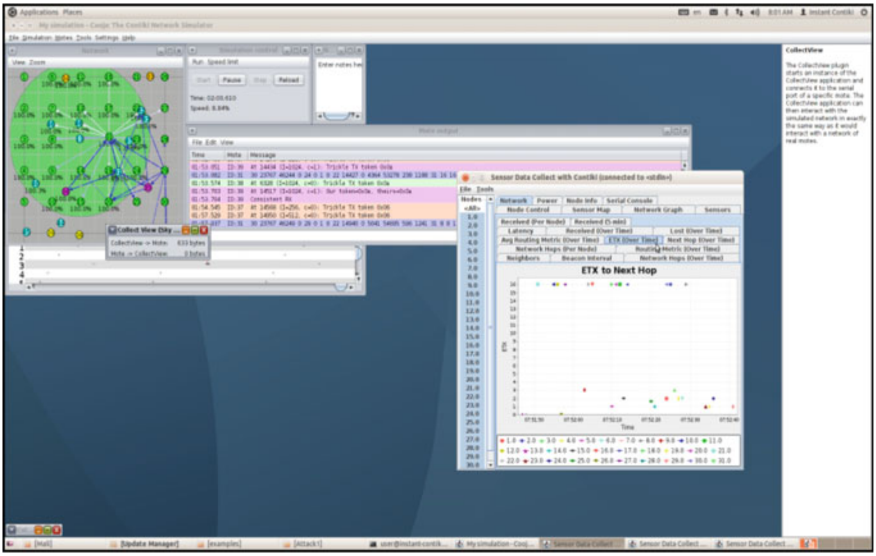
Factors compared	GNSS algorithm	Proposed meta-heuristic algorithm
Methodology	Obstacles surveyed	Fuzzy probability
Output	Less	High
Bandwidth	30%	35%
Process performance	40%	45%
Delay	40%	20%
Data security	Low	High
Loss of packet	High	Medium
Latency rate	High	Medium

7 Simulation Results

Figures 5, 6 and 7 project the proposed work performance under uncertain environment where the performance is preferable than the existing algorithms where the data transmission and hop count information and the packet received projects the high impact. The proposed algorithm projects a novel approach in optimal path selection where the proposed solution is less than 4 dB. When an UAV node is moved out of coverage from the base station, the proposed model will be able to support the optimal path amidst the base station support, thus proving the performance of the proposed algorithm.

Higher data transmission will provide better result as shown in Fig.4. The high engaged neighbor count will increase the data transmission rate where it can achieve optimal path in uncertain dynamic environment is shown in Fig.5. The data loss packet is projected in Fig. 6.

The comparisons of performance metrics of the existing algorithms and the proposed optimal algorithm are projected in Table 4. The proposed optimal algorithm applies a fuzzy possibility which provides an optimal path instantly and also is capable of overcoming the obstacles in real time. And the data security is not provided in the existing algorithms, whereas the proposed blockchain architecture mitigates the path deviation attacks and security attacks. And the proposed algorithm reduces the delay of time and also the throughput increased. The path loss is set as 150s. Loss of path values λnu is less when the UAV moves at higher height. The proposed algorithm improvements are 4.5 dB, 3.6 dB, and 2.8 dB with the UAV



```
Time       Mote   Message
01:53.051  ID:39  At 14434 (I=1024, c=1): Trickle TX token 0x0a
01:53.082  ID:31  30 23767 46244 0 24 0 1 0 22 14427 0 4364 53278 230 1188 31 16 16
01:53.574  ID:38  At 6328 (I=1024, c=0): Trickle TX token 0x0a
01:53.703  ID:38  At 14517 (I=1024, c=1): Our token=0x0a, theirs=0x0a
01:53.704  ID:39  Consistent RX
01:54.545  ID:37  At 14568 (I=256, c=0): Trickle TX token 0x06
01:57.529  ID:37  At 14050 (I=512, c=0): Trickle TX token 0x06
01:57.037  ID:31  30 23767 46240 0 26 0 1 0 22 14940 0 5041 54695 596 1241 31 8 8 1
```

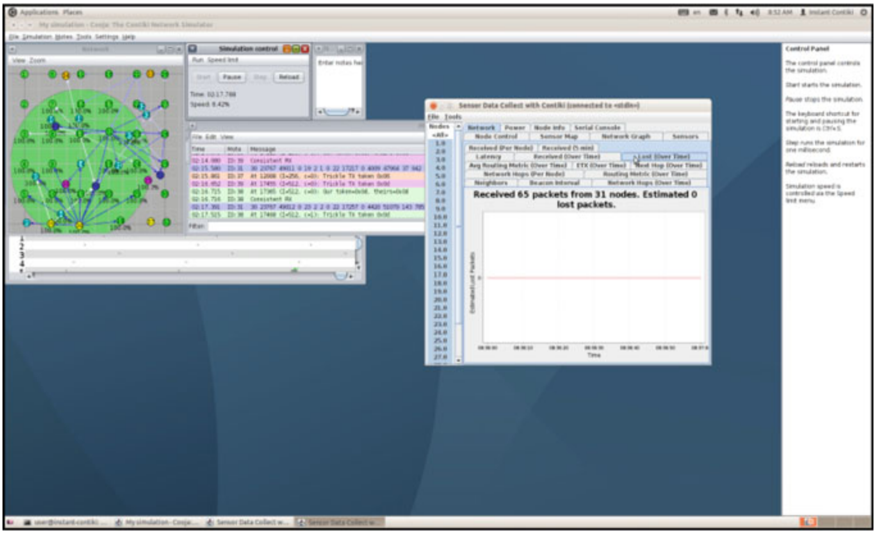
CollectView

The CollectView plugin starts an instance of the CollectView application and connects it to the serial port of a specific mote. The CollectView application can then interact with the simulated network in exactly the same way as it would interact with a network of real motes.

Time: 02:03.610
Speed: 8.84%

CollectView -> Mote: 633 bytes
Mote -> CollectView: 0 bytes

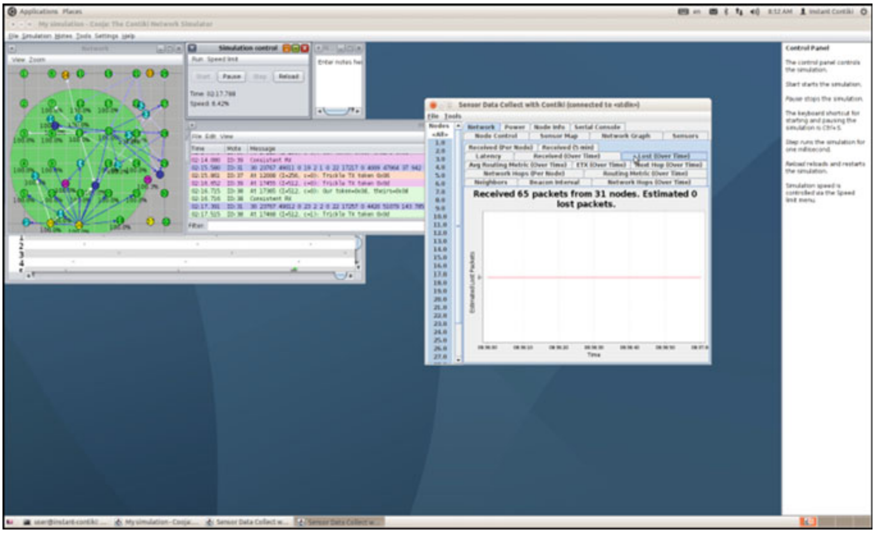
Fig. 5 Transmission data rate



```
Time       Mote   Message
02:14.980  ID:39  Consistent RX
02:15.580  ID:31  30 23767 49011 0 19 2 1 0 22 17217 0 4009 47964 37 942
02:15.861  ID:37  At 12088 (I=256, c=0): Trickle TX token 0x06
02:16.452  ID:36  At 17455 (I=512, c=0): Trickle TX token 0x0d
02:16.715  ID:38  At 17365 (I=512, c=0): Our token=0x08, theirs=0x08
02:16.716  ID:38  Consistent RX
02:17.391  ID:31  30 23767 49012 0 23 2 2 0 22 17257 0 4420 51079 143 785
02:17.525  ID:38  At 17468 (I=512, c=1): Trickle TX token 0x0d
```

Received 65 packets from 31 nodes. Estimated 0 lost packets.

Fig. 6 Neighbor count



```
Time       Mote   Message
02:14.980  ID:39  Consistent RX
02:15.580  ID:31  30 23767 49011 0 19 2 1 0 22 17217 0 4009 47964 37 942
02:15.861  ID:37  At 12088 (I=256, c=0): Trickle TX token 0x06
02:16.452  ID:36  At 17455 (I=512, c=0): Trickle TX token 0x0d
02:16.715  ID:38  At 17365 (I=512, c=0): Our token=0x08, theirs=0x08
02:16.716  ID:38  Consistent RX
02:17.391  ID:31  30 23767 49012 0 23 2 2 0 22 17257 0 4420 51079 143 785
02:17.525  ID:38  At 17468 (I=512, c=1): Trickle TX token 0x0d
```

Received 65 packets from 31 nodes. Estimated 0 lost packets.

Fig. 7 Estimated loss of packet and packet received

network flying object height as 15 m, 55 m, 120 m, and 180 m, respectively. The proposed algorithm results improvements are 4.3 dB, 3.4 dB, and 2.6 dB with the UAV network object flying height as 20 m, 60 m, 120 m, and 180 m, respectively.

8 Conclusion

In this paper, we have discussed on the issues in unmanned aerial vehicles optimal path planning and how security attacks can be mitigated. The proposed meta-heuristic optimization algorithm path planning provides instant path selection under uncertain environment where the data transmission is done securely. The proposed secure communication model will mitigate the security attacks by the blockchain distribution system. It supports routing table which is updated and scheduled and the data transmission is done. The experimental results are evaluated, and the proposed blockchain model supports ensures network security. For future works, multi-agent machine learning-based path planning is done where the energy consumption of the UAVs is studied for effective operations.

References

1. Zeng Y, Wu Q, Zhang R (2019) Accessing from the sky: a tutorial on UAV communications for 5G and beyond. In: Proceedings of the IEEE 107:2327–2375
2. Patle BK, Pandey A, Jagadeesh A, Parhi DR (2018) Path planning in uncertain environment by using firefly algorithm. Defence Technol 14(6):691–701
3. Gjanci P, Petrioli C, Basagni S, Phillips CA, Bölöni L, Turgut D (2017) Path finding for maximum value of information in multi-modal underwater wireless sensor networks. IEEE Trans Mob Comput 17(2):404–418
4. Sorbelli FB, Conti M, Pinotti CM, Rigoni G (2020) UAVs Path deviation attacks: survey and research challenges. In: 2020 IEEE international conference on sensing, communication and networking (SECON Workshops), pp 1–6. IEEE
5. Chen J, Feng Z, Wen JY, Liu B, Sha L (2019) A container-based dos attack-resilient control framework for real-time uav systems. In: 2019 design, automation & test in Europe conference & exhibition (DATE), pp 1222–1227. IEEE
6. Xu X, Zhao H, Yao H, Wang S (2020) A blockchain-enabled energy-efficient data collection system for UAV-assisted IoT. IEEE Internet Things J 8(4):2431–2443
7. Wu Y, Yang W, Guan X, Wu Q (2020) Energy-efficient trajectory design for UAV-enabled communication under malicious jamming. IEEE Wirel Commun Lett 10(2):206–210
8. Wang Y, Bai P, Liang X, Wang W, Zhang J, Fu Q (2019) Reconnaissance mission conducted by UAV swarms based on distributed PSO path planning algorithms. IEEE Access 7:105086–105099
9. Liu Q, Shi L, Sun L, Li J, Ding M, Shu F (2020) Path planning for UAV-mounted mobile edge computing with deep reinforcement learning. IEEE Trans Veh Technol 69(5):5723–5728
10. Liu G, Wang X, Liu B, Wei C, Li J (2019). Path planning for multi-rotors UAVs formation based on ant colony algorithm. In: 2019 international conference on intelligent computing, automation and systems (ICICAS), pp 520–525. IEEE
11. Jia Z, Qin X, Wang Z, Liu B (2019) Age-based path planning and data acquisition in UAV-assisted IoT networks. In: 2019 IEEE international conference on communications workshops (ICC Workshops), pp 1–6. IEEE
12. Gupta GP, Chawra VK, Dewangan S (2019) Optimal path planning for UAV using NSGA-II based metaheuristic for sensor data gathering application in wireless sensor networks. In: 2019 IEEE international conference on advanced networks and telecommunications systems (ANTS), pp 1–5. IEEE

13. Chen J, Li M, Yuan Z, Gu Q (2020). An improved a* algorithm for UAV path planning problems. In 2020 IEEE 4th information technology, networking, electronic and automation control conference (ITNEC) vol 1, pp 958–962. IEEE
14. Chawra VK, Gupta GP (2020) Multiple UAV path-planning for data collection in cluster-based wireless sensor network. In: 2020 first international conference on power, control and computing technologies (ICPC2T), pp 194–198 IEEE
15. Yoon J, Doh S, Gnawali O, Lee H (2020) Time-dependent ad-hoc routing structure for delivering delay-sensitive data using UAVs. IEEE Access 8:36322–36336
16. Binol H, Bulut E, Akkaya K, Guvenc I (2018) Time optimal multi-UAV path planning for gathering its data from roadside units. In: 2018 IEEE 88th vehicular technology conference (VTC-Fall), pp 1–5. IEEE
17. Yanmaz E, Kuschnig R, Quaritsch M, Bettstetter C, Rinner B (2011) On path planning strategies for networked unmanned aerial vehicles. In: Proceedings IEEE conference on computer communications workshops (INFOCOM WKSHPS), pp 212–216
18. Chandhar P, Larsson EG (2019) Massive MIMO for connectivity with drones: case studies and future directions. IEEE Access 7:94676–94691
19. Garcia-Rodriguez A, Geraci G, Lopez-Perez D, Giordano LG, Ding M, Bjornson E (2019) The essential guide to realizing 5G-connected UAVs with massive MIMO. IEEE Commun Magazine pp 2–8
20. Zeng Y, Lyu J, Zhang R (2019) "Cellular-connected UAV: potential, challenges, and promising technologies. IEEE Wirel Commun 26:120–127
21. Rana T, Shankar A, Sultan MK, Patan R (2019) An intelligent approach for UAV and drone privacy security using blockchain methodology 4(13). 978–1–5386–5933–5/19. IEEE
22. Chen J, Feng Z, Wen JY, Liu B, Sha L (2019) A container-based DoS attack-resilient control framework for real-time UAV systems 1(18) 978–3–9819263–2–3/DATE19/ DAA 2019
23. Xu X, Zhao H, Yao H, Wang S (2020) a blockchain-enabled energy efficient data collection system for UAV assisted IoT, pp 2327–4662. IEEE
24. Wu Y, Yang W, Guan X, Wu Q (2020) Energy-efficient trajectory design for UAV-enabled communication under malicious jamming, pp 2162–2337
25. Yang Q, Jang S-J, Yoo S-J (2020) Q-learning-based fuzzy logic for multi-objective routing algorithm in flying ad hoc networks. Wirel Pers Commun 113:115–138
26. Jobaer S, Zhang Y, Hussain MAI, Ahmed F (2020) UAV-assisted hybrid scheme for urban road safety based on VANETs. Electronics 9:1499
27. Yuan Y, Yu ZL, Gu Z, Yeboah Y, Wei W, Deng X, Li J, Li Y (2019) A novel multi-step Q-learning method to improve data efficiency for deep reinforcement learning. Knowl Based Syst 175:107–117
28. Varshosaz M, Afary A, Mojaradi B, Saadatseresht M, Parmehr EG (2020) Spoofing detection of civilian UAVs using visual odometry. Int J Geo-Inf 9(1):6
29. Liu D, Xu Y, Wang J, Xu Y, Anpalagan A, Wu Q, WangH, Shen L (2019) Self-organizing relay selection in uav communication networks: a matching game perspective. IEEE Wirel Commun
30. Lhazmir S, Oualhaj OA, Kobbane A, BenOthman J (2019) UAV for energy-efficient IoT communications: matching game approach. In: 2019 IEEE global communications conference (GLOBECOM), pp 1–6. IEEE
31. Zhou F, Hu RQ, Li Z, Wang Y (2020) Mobile edge computing in unmanned aerial vehicle networks. IEEE Wirel Commun 27(1):140–146
32. Yao H, Mai T, Wang J, Ji Z, Jiang C, Qian Y (2019) Resource trading in blockchain-based industrial internet of things. IEEE Trans Industr Inf 15(6):3602–3609

Internet of Behavior in Cybersecurity: Opportunities and Challenges

Sagar Patel and Nishant Doshi

Abstract There is a dark side to the Internet of things, and experts believe that the integration of behavioral data may supply hackers with sensitive information about consumer behavior patterns. Cybercriminals may gather and sell compromised property access codes, delivery routes, and even bank access credentials to other thieves—the possibilities are limitless. Despite many public awareness efforts, researchers discovered that people continue to use hazardous password habits. Individual variations in cybersecurity practices are the subject of little study. This review focused on the perilous habit of password sharing. As expected, people with a low level of persistence were more inclined to disclose passwords. Passwords were more likely to be disclosed by older people and those who are skilled at self-monitoring.

Keywords IoB · IoT · Cyber hygiene · Threats · Awareness

1 Introduction

As per the National Initiative for Cybersecurity Careers and Studies, information and communication systems, as well as the information they contain, are secured against and/or defended against damage caused by and/or exploitation, unauthorized use or modification, or exploitation. Security system analysts work alongside computer system users, as well as cyberattackers and the systems they are trying to get into. Unauthorized information is often the target of cyberattacks that aim to gain, change, or retain it.

Many feel that information technology developments and software development are the primary means of increasing information security, which is why the majority of cybersecurity research has concentrated on making computer networks more secure.

S. Patel (✉) · N. Doshi
Department of Computer Science and Engineering, Pandit Deendayal Energy University, Gandhinagar, Gujarat 382421, India
e-mail: sagar.pmtcs20@sot.pdpu.ac.in

N. Doshi
e-mail: nishant.doshi@sot.pdpu.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_14

As an alternative to breaking into a computer system, cyberattackers may utilize social engineering (such as fooling computer system users into providing personal information, such as passwords) and cognitive hacking (such as disseminating false information) to infiltrate a network. According to, 28% of all cybersecurity assaults are the result of social engineering, with 24% of these attacks being the result of phishing. Social engineering assaults have been effective in recent years, according to Cyber Edge Reports. Human error is the most major cybersecurity risk. Phishing (and spear-phishing) assaults, according to reports, use different approaches to fraud and to trick victims into downloading malware or visiting phony platforms in order to get their login information. Victims of these attacks are often inundated with emails and texts purporting to be notifications from a bank or social networking site, or even as part of a software update or actual communication from a third-party source. Phishing assaults are not the only cybersecurity mistakes made by computer users. These include sharing passwords with family and friends and failing to apply software updates.

In terms of adhering to security habits, computer system users range widely in their approach. People's tendency to put things off to worry about the future or to take risks has been linked to disparities in how well security measures are followed. Since human mistake may have a significant influence on network security, we will talk about psychological ways to help people follow security regulations better. The use of unique polymorphic security alerts, rewarding good cyber behavior and punishing bad, as well as raising consideration of future consequences are a few examples of psychological tactics.

To begin with, we will talk about common computer security mistakes that people make, such sharing passwords, falling for phishing scams, and forgetting to apply software updates. Third, we investigate the individual differences that underpin computer system user cybersecurity behaviors such as time management, instability, advanced anticipating, and contingency. Finally, we discuss psychological approaches for persuading users to adopt safer behaviors.

1.1 What is IoB?

In the Internet of things (IoT), actual devices are linked to gather and share data. The IoT is continually changing in terms of its complexity, i.e., how devices are connected, how autonomous things may execute calculations, and how data is stored in the cloud. Data gathering gives vital information on customer behavior, interests, and preferences (IoB). The IoB seeks to comprehend user Internet activity data from a behavioral psychology viewpoint. From a human psychology standpoint, it aims to analyze data and use that knowledge to build and promote new goods [1, 2].

The IoB method analyzes user-controlled data from a behavioral psychology viewpoint. As a consequence of such a study, new methods for user experience design, search experience optimization, and marketing of end-goods and services have been developed. So, doing IoB is straightforward technically, but complicated psychologically. It necessitates statistical studies that capture daily routines and behaviors without totally compromising customer privacy.

Also, IoB integrates technologies that directly target individuals, like face recognition, location monitoring, and big data. It combines three disciplines: technology, data analytics, and behavioral psychology.

1.2 The Benefits of IoB?

- IoB can assist companies in resolving problems related to increasing sales and maintaining a high level of customer satisfaction.
- It can even be used to replace numerous customer surveys, which are time intensive for both consumers and enterprises.
- IoB enables you to evaluate your prospects' activity and purchasing patterns across many platforms.
- You may get previously unavailable data on how prospective consumers engage with your company, goods, and services.
- You may get a deeper knowledge of your customer's purchasing habits.
- This will also enable you to notify your consumers in real time about any new offers, points of sale, or even targeted advertisements [1–3].

2 Factors of Internet of Behaviors

2.1 Changing Value Chains

The Internet of behavior is a trinity of technology, data analytics, and behavioral science that enables us to forecast, analyze, and even affect human behavior. Simultaneously, IoB technologies have the potential to transform the value chains that connect platforms and people. While some technologically savvy individuals are opposed to freely sharing their personal information, the majority of consumers are satisfied as long as such platforms provide value or make their lives simpler [3].

Businesses take advantage of this potential by offering services that amass massive quantities of data that may ultimately be used to alter behavior [4, 5]. While the value offer may seem enticing, individuals' liberty is jeopardized.

2.2 Privacy Concerns

The more data generated by IoT, the more efficient IoB algorithms are in predicting individual behavior. Numerous businesses acquire and sell data. PayPal, for example, has revealed that it distributes customer data with hundreds of organizations worldwide, including name, address, and phone number [6]. Additionally, monopolies like as Uber continue to purchase rival applications that aggregate user data, often without the consumers' consent. This poses serious legal and security concerns to individuals' privacy rights.

Continuous observation of people through IoB entails a surveillance risk. While there are distinctions between government data collection and private company data collection, governments may always compel private businesses to provide data or sell data to government organizations [7]. For example, the Indian Government is preparing to introduce new laws requiring social media companies to provide information voluntarily.

2.3 Security Threats

With the worldwide increase in cybersecurity risks, hackers gaining access to IoT data combined with behavioral data may pose serious personal security concerns. This may result in hackers obtaining information such as access codes to properties, delivery routes, and even bank access codes.

Phishing is a phenomenon in which a criminal impersonates another person in order to trick people into disclosing sensitive information through different digital media. With access to behavioral data, an attacker may take phishing to a whole new level by impersonating people in order to commit fraud or other malicious acts.

3 Comparative Work

The whole section discusses the several types of cybersecurity blunders that computer system users make. Numerous assessments have shown that people provide the greatest security risk, and further research has verified this [1]. Human mistakes, according to one study, are responsible for 95% of cyber and network attacks. Individuals are categorized as system users or security professionals in this context, with the bulk of research focusing on user errors. Coworkers are the biggest liability in a company's security chain [1] (Table 1).

These are only a few instances of careless mistakes in cyber and network-related security. Increasing network or data availability and accessibility while preserving security is a key problem in information and cybersecurity [15]. Organizations typically require complex passwords for computer users to increase security, which hinders usability. Computer users choose the easy approach, such as using a weak password on several Websites. Here are three examples of human security failures: phishing, password sharing, and not updating software.

Falling victim to phishing: Experimental phishing research has been employed in certain phishing research. A recent study has shown that the use of laboratory-based phishing trials correlates with real-world phishing [15, 16]. According to one survey, over 30% of government workers click on a suspicious link in this phishing email, and a significant number of these employees disclose their credentials. Another study found that almost 60% of university students clicked on questionable links in a phishing email [1].

Sharing passwords: Password sharing with friends, family, and even strangers is a frequent occurrence of human cybersecurity mistakes. Perseverance and self-monitoring are associated with a greater likelihood of older people sharing passwords. Password sharing may result in the financial exploitation of elderly individuals, one of the most prevalent types of maltreatment [3]. This is because many older people

Table 1 Comparative work

Paper title and year	Factors affecting	Methodology	Conclusion
Cybersecurity awareness, knowledge, and behavior: a comparative study [8]	Cyber awareness; cyberthreats; cybersecurity; cyber behavior; cyber knowledge;	Emphasized the need of using a comparative approach when evaluating cultural variations in cybersecurity awareness, knowledge, and behavior. However, future research should focus on the underlying causes of a lack of cyberthreat awareness	The implications of this research and suggestions for successful, evidence-based cybersecurity training programs are provided and addressed in this article [8]
A pilot study of cybersecurity and privacy-related behavior and personality traits [9]	Phishing, security, human factors, privacy, personality traits, facebook	Phishing is a kind of attack that involves the use of false electronic mail (email) that seems to be from a reputable source. Phishing emails are designed to elicit personal information from users, such as user IDs and passwords – Neuroticism – Extroversion – Openness – Agreeableness – Conscientiousness	The study investigates the characteristics that may lead to an individual's vulnerability to online security and privacy assaults. The purpose of this research is to determine the relationship between personality factors and phishing email response[9]

(continued)

Table 1 (continued)

Paper title and year	Factors affecting	Methodology	Conclusion
An exploratory study of cyber hygiene behaviors and knowledge [10]	Age, cyber hygiene, cybersecurity,	They have prepared the survey of the cyber hygiene among different age groups and concluded their knowledge and prepared a detailed survey on it. [10]	These survey and findings related to the cybersecurity can be used for the improving the cyber hygiene among the people. It also helps to reduce the cybercrimes and make aware the people about it [10]
Correlating human traits and cybersecurity behavior intentions [11]	Cybersecurity behaviors, cybersecurity intentions surveys, human factors individual differences	Four predictor variables were evaluated, each reflecting one of the four primary areas of individual differences: demographic characteristics, personality traits, risk-taking inclinations, and decision-making styles. The study findings regarding security practices such as device encryption, password generating, proactive awareness, and upgrading [11]	By integrating demographic characteristics and personality traits in our survey, we extend Egelman and Peer's findings on the effect of individual variations on cybersecurity behavior intentions [11]
Internet of behaviors (IoB) and its role in customer services [12]	Internet of things (IoT), Industry 4.0, Internet of behavior (IoB), applications, customer behavior,	Analyze human activities, change the culture, customer habit, monitoring, track buying habits	The data may serve as the foundation for a business's growth, marketing, and sales strategy. Numerous new data and materials may be analyzed by industries. Additionally, it contributes to the development of profit and customer happiness. The IoB contributes to journey research by collecting data from multiple touchpoints. This results in the creation of additional points and new channels of communication with customers [12]

(continued)

Table 1 (continued)

Paper title and year	Factors affecting	Methodology	Conclusion
The role of user behavior in improving cybersecurity management [13]	Phishing, cognitive hacking, cybersecurity, social engineering	Rewarding and punishing appropriate and inappropriate cyber activity; increasing consideration for the long-term consequences of acts [13]	According to the study, certain behavioral patterns such as hyperactivity, recklessness, and an incapacity to evaluate the long-term consequences of actions are connected with non-compliance with cyber and network security procedures [13]
Leveraging behavioral science to mitigate cyber security risk [14]	Cybersecurity, heuristics risk, communication health models, cognitive load, bias,	The empirical assessment of the consequences of change on cybersecurity entails a number of steps, including identifying factors, correcting for bias and interaction effects, and assessing the generalizability of conclusions. These are important concepts of the empirical method, yet they are often misunderstood or misapplied [14]	It will serve as a guide for determining when and how behavioral concerns should be included into the layout, specification, execution, and usage of cybersecurity products and procedures. [14]

have a high level of trust in others and strangers, particularly on the Internet. As with older folks, younger adults exchange passwords, particularly those for streaming services [1]. The majority of us share our passwords across several Websites, and sharing a password enables outsiders to access other private information. Sharing passwords is inherently dangerous since fraudsters may use them on a variety of different Websites once they find them on one system [17].

Installing software updates: One of the most pervasive issues behind cybersecurity practices is a failure to implement software upgrades on time or at all [18–20]. They discovered that risk-taking behaviors may partly explain for some individuals' behavior when it comes to software security patches, including some who are more risk-averse choose to delay software updates installation. In compared to credential information exchange and spoofing, the industry has paid less attention to technical improvements [13].

4 Challenges of IoB

- Hackers may use behavioral data to get access to sensitive information about customer trends, collect and sell property access codes, delivery routes, and banking credentials [3].
- Because the degree of access to IoB is difficult to monitor, all businesses must be aware of the risks associated with its usage.
- Data theft and information breaches may be very damaging, which is why new cybersecurity procedures are needed to maintain openness and client privacy.
- Data must be used transparently because it enables individuals with behavioral science understanding to properly control their behavior. One method for securing data is to require corporations to get permission before collecting personal data, to delete biometric identifiers if required, and to securely store information.

5 Conclusion

Our research may be beneficial to system designers who are developing systems that need secure functioning. To protect the safety of children of all ages, irrespective of their level of expertise, we must provide unambiguous security signs and improved instruction. Additionally, we must dissect the sorts of retraining that user include in order to ascertain why cyberspace training seems to be ineffective. Users claim that the bulk of their awareness of correct behavior comes from public sources, such as technical experts, family members and friends, or perhaps a local computer store or retailer. Investigators should assess the reliability and utility of various sources of information.

References

1. Fatokun Faith B, Hamid S, Norman A, Fatokun Johnson O, Eke CI (March 2020) Relating factors of tertiary institution students. Cybersecurity Behav. https://doi.org/10.1109/ICMCECS47690.2020.246990
2. Li S, Feng B, Liao W, Pan W (1 June 2020) Internet use, risk awareness, and demographic characteristics associated with engagement in preventive behaviors and testing: cross-sectional survey on COVID-19 in the United States. J Med Int Res 22(6). JMIR Publications Inc. https://doi.org/10.2196/19782
3. de Kimpe L, Walrave M, Verdegem P, Ponnet K (2021) What we think we know about cybersecurity: an investigation of the relationship between perceived knowledge, internet trust, and protection motivation in a cybercrime context. Behav Inf Technol.https://doi.org/10.1080/0144929X.2021.1905066
4. Quayyum F, Cruzes DS, Jaccheri L (1 Dec 2021) Cybersecurity awareness for children: a systematic literature review. Int J Child-Comput Inter 30. Elsevier B.V. https://doi.org/10.1016/j.ijcci.2021.100343

5. Gillam AR, Foster WT (July 2020) Factors affecting risky cybersecurity behaviors by U.S. workers: an exploratory study. Comput Hum Behav 108. https://doi.org/10.1016/j.chb.2020.106319
6. Lee JK, Chang Y, Kwon HY, Kim B (2020) Reconciliation of privacy with preventive cybersecurity: the bright internet approach. Inf Syst Front 22(1):45–57. https://doi.org/10.1007/s10796-020-09984-5
7. Venard B (2021) Cybersecurity behavior under covid-19 influence. In: 2021 international conference on cyber situational awareness, data analytics and assessment (CyberSA), pp 1–9. https://doi.org/10.1109/CyberSA52016.2021.9478238
8. Zwilling M, Klien G, Lesjak D, Wiechetek Ł, Cetin F, Basim HN (2020) Cyber security awareness, knowledge and behavior: a comparative study. J Comput Inf Syst 00(00):1–16. https://doi.org/10.1080/08874417.2020.1712269
9. Halevi T, Lewis J, Memon N (2013) A pilot study of cyber security and privacy related behavior and personality traits. WWW 2013 companion—proceedings of the 22nd international conference on world wide web, pp 737–744. https://doi.org/10.1145/2487788.2488034
10. Cain AA, Edwards ME, Still JD (2018) An exploratory study of cyber hygiene behaviors and knowledge. J Inf Sec Appl 42:36–45. https://doi.org/10.1016/j.jisa.2018.08.002
11. Gratian M, Bandi S, Cukier M, Dykstra J, Ginther A (2018) Correlating human traits and cyber security behavior intentions. Comput Secur 73:345–358. https://doi.org/10.1016/j.cose.2017.11.015
12. Javaid M, Haleem A, Singh RP, Rab S, Suman R (2021) Internet of Behaviours (IoB) and its role in customer services. Sens Int 2(7):100122. https://doi.org/10.1016/j.sintl.2021.100122
13. Moustafa AA, Bello A, Maurushat A (2021) The role of user behaviour in improving cyber security management. Front Psychol 12(June):1–9. https://doi.org/10.3389/fpsyg.2021.561011
14. Pfleeger SL, Caputo DD (2012) Leveraging behavioral science to mitigate cyber security risk. Comput Secur 31(4):597–611. https://doi.org/10.1016/j.cose.2011.12.010
15. Grobler M, Gaire R, Nepal S (March 2021) User, usage and usability: redefining human centric cyber security. Front Big Data 4. https://doi.org/10.3389/fdata.2021.583723
16. Mashiane T, Kritzinger E (2019) Cybersecurity behaviour: a conceptual taxonomy. In: Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics), vol 11469 LNCS, pp 147–156. https://doi.org/10.1007/978-3-030-20074-9_11
17. Sechi C, Loi G, Cabras C (2021) Addictive internet behaviors: the role of trait emotional intelligence, self-esteem, age, and gender. Scand J Psychol 62(3):409–417. https://doi.org/10.1111/sjop.12698
18. Critselis E et al (2014) Predictive factors and psychosocial effects of internet addictive behaviors in cypriot adolescents. Int J Adolesc Med Health 26(3):369–375. https://doi.org/10.1515/ijamh-2013-0313
19. Whitty M, Doodson J, Creese S, Hodges D (2015) Individual differences in cyber security behaviors: an examination of who is sharing passwords. Cyberpsychol Behav Soc Netw 18(1):3–7. https://doi.org/10.1089/cyber.2014.0179
20. Arend I, Shabtai A, Idan T, Keinan R, Bereby-Meyer Y (2020) Passive- and not active-risk tendencies predict cyber security behavior. Comput Secur 97:101964. https://doi.org/10.1016/j.cose.2020.101964

Fusion of Federated Learning and 6G in Internet-of-Medical-Things: Architecture, Case Study and Emerging Directions

Ashwin Verma, Pronaya Bhattacharya, Ishan Budhiraja, Amit Kumar Gupta, and Sudeep Tanwar

Abstract Smart health care has transitioned Internet-of-Medical-Things (IoMT) to share data via sensor nodes among a large number of users. Earlier, IoMT data was mainly stored on centralized cloud servers for analytics. Cloud server induces challenges of high-end computational latency, network bottlenecks and privacy and security issues against adversarial attacks. Thus, the paradigm has shifted towards decentralized learning through the deployment of tiny models with local data via federated learning (FL) schemes. Coupled with sixth-generation (6G) network services, humongous sensor data exchange is possible in real time. Thus, 6G-assisted FL addresses the open challenges of responsive network orchestration and secured user privacy. Thus, the paper proposes a survey article on the fusion of FL and 6G at the backdrop of IoMT ecosystems. A 6G-assisted FL reference architecture for IoMT is proposed, and an (ϵ, δ) differential privacy preserving 6G-assisted FL case study is proposed for IoMT. We also discuss the potential open issues and future research directions on the integration of FL and 6G in IoMT. The survey intends to assist researchers, and healthcare stakeholders to design secure FL schemes for IoMT ecosystems assisted with 6G networks.

A. Verma (✉) · P. Bhattacharya · S. Tanwar
Institute of Technology, Nirma University, Ahmedabad, Gujarat, India
e-mail: ashwin.verma@nirmauni.ac.in

P. Bhattacharya
e-mail: pronoya.bhattacharya@nirmauni.ac.in

S. Tanwar
e-mail: sudeep.tanwar@nirmauni.ac.in

A. Verma · A. K. Gupta
Amity School of Engineering and Technology, Amity University, Jaipur, Rajasthan, India
e-mail: dramitkumargupta193@gmail.com

I. Budhiraja
School of Engineering and Applied Sciences, Bennett University, Greater Noida, Uttar Pradesh, India
e-mail: ishan.budhiraja@bennett.edu.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_15

Keywords 6G · Enhanced reliable low-latency communications · Federated learning · Internet-of-Medical-Things

1 Introduction

Internet-of-Medical-Things (IoMT) assists a sensor-driven network of medical nodes (wearables) that are placed strategically and are used to measure various attributes like heart rate, pulse, pressure, brain signalling and many others. In IoMT, the collected sensor data is sent to cloud servers for AI-based analytics [1]. However, due to the limitations of high-bottlenecks, single-point failures and security attacks, the focus has shifted towards decentralized analytics in IoMT. In decentralized analytics, the patient data generated by sensor nodes is required to be analysed at different points in the network, which might lead to issues of data fragmentation, control and data redundancy. As per the reports by *RBC Capital Markets*, by 2025, 36% growth in healthcare data is expected, which is approx 9.3% faster than other allied sectors like manufacturing, finance and infotainment [2]. Centralized servers materialize the data in different views of end-users, based on associated query [3]. However with strong regulation scenarios and privacy concerns, patient data must be shared with trusted and authorized stakeholders. Thus, due to the dual challenges of high data ingestion and user data privacy in decentralized IoMT, AI-based analytics has shifted to the edge nodes, where tiny deployed models work on local patient data.

To support edge-based networking, researchers have shifted towards mobile edge networks, that can support low latency among edge nodes via fifth-generation (5G) ultra-reliable low-latency communication services (uRLLC) [4]. However, with the explosion of sensor-generated data in the near future, current 5G-uRLLC networks would face bottlenecks to address real-time communication requirements in IoMT. Thus, with the advent of sixth-generation (6G)-enhanced reliable low-latency communication (ERLLC) networks, the 5G-uRLLC service is improved many folds to address round trip time (RTT) latency constraints. 6G-ERLLC supports an RTT of 0.1 ms, with support of edge-intelligence at low power due to intelligent networked stack. Thus, the inclusion of 6G networks in IoMT ecosystems to address latency requirements is a viable option [5]. To realize this on 6G, we need sub-Terahertz and Terahertz spectrum which provide higher frequency against millimetre spectrum used in 5G.

However, the vision of effective analytic's in IoMT is still far from reality. Healthcare data is collected from heterogeneous wearable and clinical institutions and contain sensitive patient personal indicators. The traditional system requires centralization of data on server machine, where all the data collected locally sent to the server for processing and sending back to the devices which limits the real-time ability to learn the model. As most machine learning (ML) and deep learning (DL) algorithms require large data aggregation from different sources, there are chances of privacy leaks. In such cases, federated learning (FL) is a viable fit in IoMT as it trains the global model, with the device local data itself [6]. As depicted in Fig. 1, in FL model

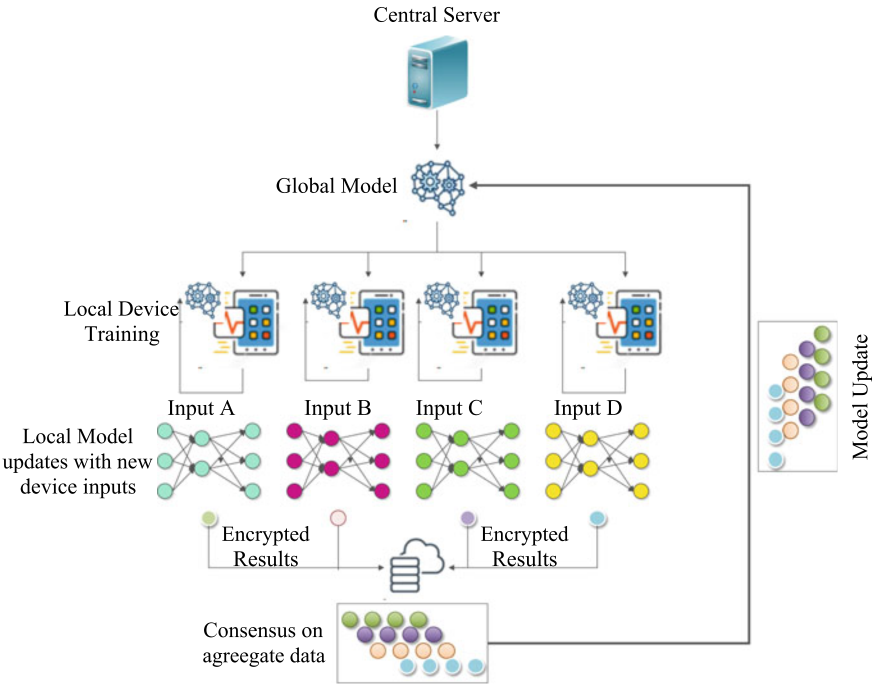


Fig. 1 Overview of FL training

training, the central model presents a globally trained model, which is copied by local device nodes. The local nodes train the model, the results are encrypted and sent back to the global model for updation [7]. Thus, FL in IoMT addresses the issue of privacy preservation and confidentiality of real-time collected patient data. Thus, the data at global models are improvised with local sensor data. All the training data remains at the IoMT devices, and no private information is communicated. However, once the FL data is collected, a resilient low-latency communication backbone is required to update the global model in real time. Thus, the fusion of 6G in FL model training in IoMT allows real-time local updates to global models. Moreover, 6G employs an intelligent low-powered communication stack, and thus the overall energy requirements in data exchange are significantly reduced. This improves the convergence rate of FL model training, as it requires fewer iterations to train the global model at lower communication costs and also preserves the privacy and anonymity of data exchange.

1.1 Novelty of the Proposed Survey

IoMT ecosystems are sensor-driven, where massive data is exchanged among open channels. Thus, to leverage real-time analytics, researchers have proposed FL-based surveys in healthcare-related to predictive models [8], edge-assisted FL multimodal analysis [9] and deep FL for sustainable health care [6]. In a similar direction, researchers have proposed 6G-assisted real-time IoMT monitoring solutions to address the issues of bandwidth and latency, with intelligent edge-AI orchestration [5]. However, IoMT requires an integrative end-to-end solution that integrates networking management elements with privacy preservation via FL. Owing to the survey gap, the paper presents a high-level architecture of 6G-assisted FL-learning model architecture based on 6G-ERLLC service that addresses the ultra-low-latency requirements of sensor communication in IoMT. As 6G models put an assertive upper bound of RTT latencies, the local models communicate the local gradients much faster to the global model, and the learning rate improves with assured privacy of data exchange.

1.2 Survey Contributions

The following are the survey contributions.

1. A 6G-envisioned FL-based IoMT architecture is presented that proposes global updation of models from locally trained IoMT sensor nodes.
2. A case study of fusion of 6G and FL is proposed, where we propose a differential privacy preserving solution that minimizes generated noise among the local trained models, to improve the gradient values, based on the (ϵ, δ)-differential privacy $(\epsilon, \delta) - DP$ approach. The case study depicts the design of the global $(\epsilon, \delta) - DP$, and denoising of the local gradients to improve the learning rate of the global model.
3. Open issues and research challenges of the practical deployments on the fusion of 6G and FL in IoMT ecosystems are discussed along with potential research directions.

1.3 Article Structure

The paper is divided into sections. Section 2 presents the related work. Section 3 presents the 6G-envisioned FL architecture in IoMT. Section 4 presents the $(\epsilon, \delta) - DP$ FL case study. Section 5 discusses the open issues and potential research directions, and finally, Sect. 6 concludes the article.

2 Related Work

Researchers have presented FL and 6G-assisted solutions to ensure privacy, security and latency requirements in healthcare applications. For example, Abdul Rahman et al. [10] presented a FL model that trains the first layer of neural network in a federated way and other layers with local data directly from the source sensors. And this combination shows the accuracy equivalent to a fully centralized model, which surpasses the latency and amount of data sent to a centralized aggregator node. Huang et al. [11] proposed predicted hospital stay time and mortality of the patient. Based on the local data of the patients at each hospital, a local learning model is created and is shared with all stakeholders, where a cluster learns and adapts the changes of each ML model. Table 1 discusses the similar approaches along with necessary parameters.

3 A View of 6G-Envisioned FL-Based IoMT Architecture

In this section, we present the reference architecture of the 6G-envisioned FL IoMT ecosystem. Figure 2 presents the details. In the proposed system, we consider that real-world data is generated via different healthcare stakeholders like hospitals, clinics and patients through the assisted wearable. The collected data is then segregated into a real-time analytical model and static models that take inputs as electronic health records (EHRs). A shared global model G_s is proposed and is shared with central server. G_s is prepared through heterogeneous local sources as aforementioned. In the reference architecture, we assume there are N healthcare clients, denoted as $\{C_1, C_2, \dots, C_n\}$. The client nodes operate on local stored data $D = \{D_1, D_2, \dots, D_n\}$ that resides on mobile gateways or even IoMT wearable. For any C_k, where $k \subset n$, we associate the data as D_k, and ω_k is considered as sample size for any D_k. The total sample size is denoted by $\sum_{k=1}^{K} \omega_k$.

The learning model parameter is denoted as w, and in FL the collected n samples are trained locally, so that privacy risks are minimized. We present the training cost as $\min_{w \in M^d} F(w) = \sum_{k=1}^{K} \frac{\omega_k}{n} F_k(w)$, where $F_k(w) = \frac{1}{\omega_k} \sum_{x_i \in D_k} f_i(w)$. The function f_i denotes the loss function, and it depends on the input-to-output conditions presented to the model. We consider $x_i \in M^d$ as the model description. Once the local training is completed, the model distributes the data statistically among N client sources, with minimum differences. To minimize the statistical differences, the federated average (FedAvg) model is generally proposed. This minimizes the skewness of data distributions. The model is trained on local data D_k, and the results are collected and communicated back to the global model via 6G-uplink and downlink channels.

For K clients, the communication bandwidth of the uplink and downlink channel is presented as $B_{\text{UL/DL}} \in O(T_{C(U)} \times |w| \times (H(\Delta_w^{\text{UL/DL}}) + \beta)$, where $T_{C(U)}$ denotes the number of updates each client node presents to the global model, $|w|$ denotes the model size, and $H(\Delta_w^{\text{UL/DL}})$ denotes the entropy of weights, and β denotes the statistical difference between true update size against the minimum update size, which is communicated by n clients. To minimize the entropy H, the value of β is also minimized, and $\min(H(\Delta_w^{\text{UL/DL}})$ communication latency is minimized. Thus, the 6G-ERLLC service provides a viable fit in such cases as $|w|$ updates are communicated faster, and thus model updates are converged in less time, which minimizes the entropy of the global model.

Table 1 Relative comparison of proposed scheme with state-of-the art approaches

Author	Year	a	b	c	d	Advantages	Limitations
Jiang et al. [12]	2020	N	–	N	Y	Proposed a comprehensive discussion of different methods of FL, and its applicability in smart cities that ensure privacy and security of the personal data	Author did not discuss the communication model used to send data to server node
Li et al. [13]	2020	N	Y	N	Y	A detailed survey on the applicability of the FL and its challenges along with opportunities is discussed	The network communication model is not discussed and entities communicate with each other
Choudhury et al. [14]	2020	N	Y	N	Y	Author discussed 2-level privacy for distributed health data with the help of FL	The communication model is not discussed that ensure the transmission of data from local node to global node
Yuan et al. [15]	2020	N	-	N	Y	Author discussed a FL-based framework to train deep neural network	Privacy of the patient’s data and how local model communicate with global server not discussed
Yin et al. [16]	2021	N	Y	N	Y	Author discussed generic privacy preserving approaches in FL and provided challenges and future directions for researchers and academician	Did not mentioned the communication model on which the data is sent to preserve the privacy from adversarial attack
Qayyum et al. [9]	2021	N	N	N	Y	Author discussed the collaborative FL approach for diagnosis of Novel Coronavirus-2019 and suggested tools and techniques for deploying it on edge devices	Privacy preservation of patients data as well the communication network to send the update is not discussed
Mothukuri et al. [17]	2021	N	Y	N	Y	Discussed overview and classification of different approaches in FL domain and provided future directions to enhance privacy and security of the data	The communication model between edge nodes and centralized cloud server is not discussed
Gadekallu et al. [18]	2021	N	Y	N	Y	The article presents a comprehensive survey on big data and its application, and how we apply FL in different domains such as health care and IoT environment	How access point is communicating with aggregator node is not discussed
Proposed	2021	Y	Y	Y	Y	Provide a detailed overview and applicability of FL in integration of 6G in health care	–

Parameters
[a] 5G/6G
[b] Privacy preservation
[c] Case study/Use-case
[d] FL
Y-denotes the parameter is considered
N- denotes the parameter is not considered

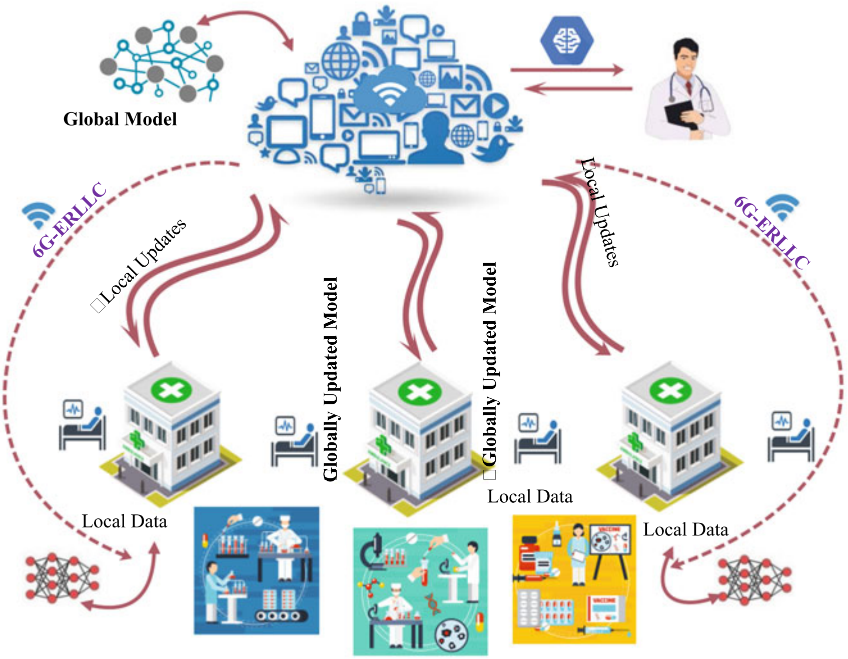


Fig. 2 Reference FL-assisted IoMT architecture

4 Case Study: (ϵ, δ)-differential Privacy Preserving 6G-assisted FL Training in IoMT

In this section, we discuss the reference architecture of the $(\epsilon, \delta) - DP$ FL scheme in IoMT ecosystems. Figure 3 presents the layered architecture of the proposed case study.

4.1 Local IoMT Training Layer

We consider k healthcare IoMT setups, denoted as $\{N_1, N_2, \ldots, N_k\}$. In any N_k, a single patient entity E_p node is setup with q sensors attached to patient as wearable, denoted as $\{S_1, S_2, \ldots, S_q\}$. The q sensors generate data $\{D_1, D_2, \ldots, D_q\}$, and the generated data D from q sensors, are mapped with E_p, through a mapping function $M : E_p \rightarrow D$. The same process is considered $\forall$ N_k IoMT setups. Thus, to refer to any associated data in kth IoMT, we represent the local data as D^k_q, that associates qth sensor in kth setup for E_p. To present in simple manner, D^k_q over all IoMT setups are presented as $\{D_1, D_2, \ldots, D_k\}$, where the sensor mapping is not considered.

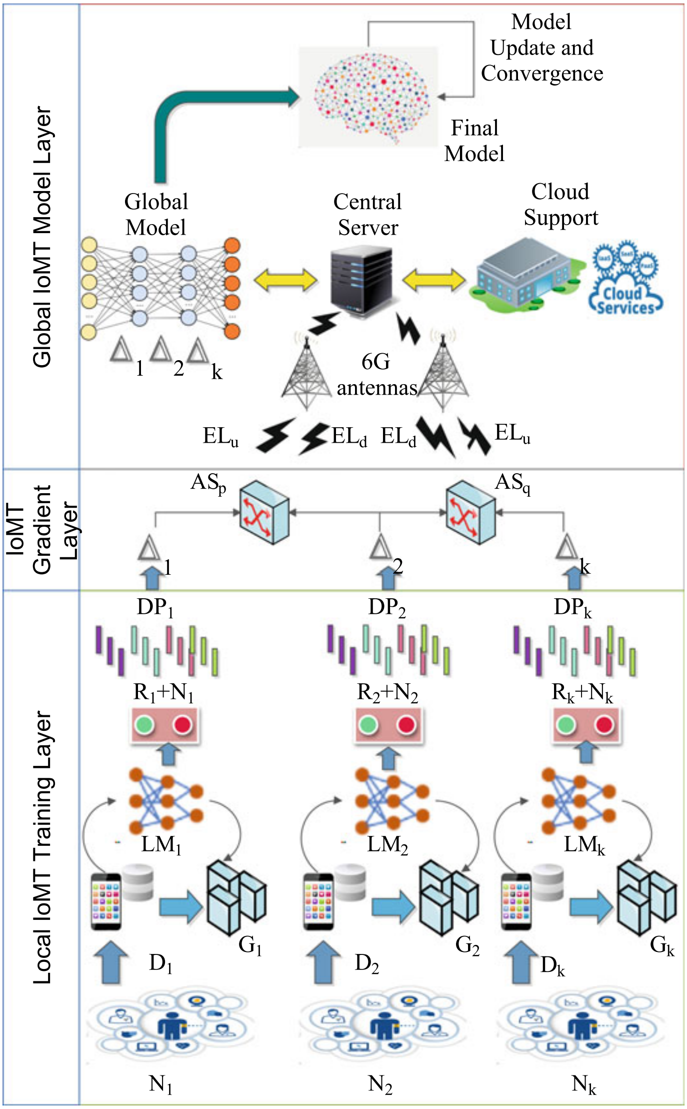


Fig. 3 Layered architecture of the proposed case study

The data is stored on local mobiles, and learning models $\{LM_1, LM_2, \ldots, LM_k\}$ are built that corresponds to the associated data. Now, the task for E_p is to associate a vector gradient G_k to D_k, based on model training with D_k. This means we have to compute vector V_k that minimizes the loss function of LM_k. The IoMT sink sensors nodes aggregate the data from remaining $(q - 1)$ nodes over all D_k patients and form an updated weight vector given as $W_t = \sum_{i=1}^{q} q_i.w_i$, where w_i is the parameter vector for ith local trained data, and $P_i = \frac{|D_q^i|}{D} \geq 0$, with the condition $\sum_{i=1}^{q} P_i = 1$. $|D|$ denotes the total size of all training data samples, with the optimization condition $w_{opt} = \arg\min_w \sum_{i=1}^{q} F_i(W)$. In this case, $F(i)$ represents the losses occurred during the training of local data D for any qth E_p.

Once w_{opt} condition is formulated, we consider the data collected from $\{LM_1, LM_2, \ldots, LM_k\}$, and apply model training to generate results $\{R_1, R_2, \ldots, R_k\}$. Once the results are generated, we assign noise samples $\{N_1, N_2, \ldots, N_k\}$ so that possible intruders cannot gain sensitive and private information of the training results. To add the noise samples, we consider the samples are collected from noise distribution $N(0, \sigma^2)$. This preserves the privacy of user samples over distributed local nodes. We consider the $\epsilon - DP$ approach, where ϵ denotes the bound. The noise is accumulated from the neighbouring data sets D_{nd} defined within a γ factor. γ represents the overlapping of the two data sets, under the bound e^{ϵ}. Thus, higher values of ϵ would result in high risk of privacy-based attacks. The Gaussian noise scaling factor is represented as $\eta \geq q\frac{\delta r}{\epsilon}$, with the constraint $q \geq \sqrt{2ln(1.25/\Delta)}$.

Under the assumptions, we define the $(\epsilon, \delta) - DP$ model for random sets $A \subseteq D$, with the domain X and range R. From A, we consider any two adjacent data sets $A_1, A_2 \in A$. The $(\epsilon, \delta) - DP$ condition is formulated as $Pr[M(A_1) \in A] \subseteq e^{\epsilon} Pr[M(A_2) \in A] + \delta$. In this case, the noise sensitivity is defined as $\delta_n = max_{A_1, A_2}||n(A_1) - n(A_2)||$. With the addition of (ϵ, δ) noise to $\{R_1, R_2, \ldots, R_k\}$, we represent the result as $R_i + N_i$, where $i \in [1, k]$. The final result $\{DP_1, DP_2, \ldots, DP_k\}$ is passed to aggregator nodes AS_p, and AS_q, which connects to the 6G-ERLLC base station nodes via the uplink EL_u.

4.2 IoMT Gradient Layer

At this layer, privacy preserved data DP_k is forwarded via AS_p and AS_q through ERLLC EL_u uplink channel. Via 6G support, the sum-rate capacity of EL_u is maximized with support of nearby non-interfering HetNets channels. Based on channel modulation, the uplink and downlink rate is monitored between different links, which reduces latency and provides high QoS. The end-to-end (EE) latency of 6G free-space optical (FSO) links is $\approx < 100ns$ over-the-air (OTA) interface. Thus, 6G allows effective real-time communication support to the main central server. Once data reaches the central server, we run the global DP-based FL updates to synchronize the main model.

4.3 Global IoMT Model Layer

At this layer, we aggregate the collected trained data, along with the gradient samples $\Delta = \{\Delta_1, \Delta_2, \ldots, \Delta_k\}$ and pass them to the central server $C(S)$ for model updates.

$C(S)$ is supported by cloud servers for resource management and task offloading in case of peak load. The gradient Δ, which follows the defined constraints by $(\epsilon, \delta) - DP$, is passed with a clip mechanism, that ensures that $||\Delta_q|| \leq C$, where $||\Delta_q||$ denotes the gradient parameters for qth E_p, and C denotes the size of the clipping window (in bytes). C is estimated based on peak arrival loads at $C(S)$ and is pre-computed through intelligent channel sensing mechanism. The batch training process $\forall D_i$, $1 \leq i \leq q$ is represented as $\tau_i^{D_i} \triangleq \arg\min_w F_i(w, D_i) = \frac{1}{|D_i|} \sum_{i=1}^{|D_q|} \arg\min_w F_i(w, D_{i,j})$, where $D_{i,j}$ is the jth sensor upload for ith E_q. The sensitivity of the global model $G(M)$ is expressed as $\Delta_{S_u}^{D_i} = \max_{D_i, D_i^{;}} ||S_u^{D_i} - S_u^{D_i^{;}}||$, where $D_i^{;}$ is nearby data set to D_i within ϵ bound. For uplink channels, the global sensitivity is updated. The final converged model $C(M)$ is communicated back via 6G-downlink El_d to local N_k nodes for re-training. With their local data, the updated model is re-trained, and the process is iterated multiple times until the final model $F(M)$ is prepared. The sent model to local IoMT nodes is represented as $S_D^{C(M)} \triangleq \sum_{i=1}^{q} w_i.S_i$. As 6G channels are used for communication, it allows resilient and real-time communication of model and gradient parameters across nodes that improves the learning rate of the ecosystem.

5 Open Issues and Potential Research Directions

In this section, we present the open issues and challenges, along with the future research directions. Figure 4 discusses the details.

5.1 FL-Based Solutions in IoMT and Potential Research Directions

Although FL-based solutions in IoMT ensures data privacy, they are challenged on different fronts, which hinders its real-time adoption as of yet.

1. *Imbalance and Skewing of data, classes and features*—In FL, the biggest problem is data imbalance, and data is highly skewed to favour a particular data class. Thus, in such cases, the training model results are often biased towards the particular class. The conditions occur when a particular node only updates the model with local data, and other nodes are not so frequent in updations. A possible solution could be to design balancing models that assure all local nodes process and train the data models. Similar problems are present with missing classes and features where one site has fewer training samples, and another site has more samples. In such cases, the resultant models are poor in design. A solution approach is again to balance the sample distribution equally among all local site nodes.
2. *Communication Power Requirements*- In IoMT, there are massive sensor nodes, and thus, communication power requirements are high. The designed FL training

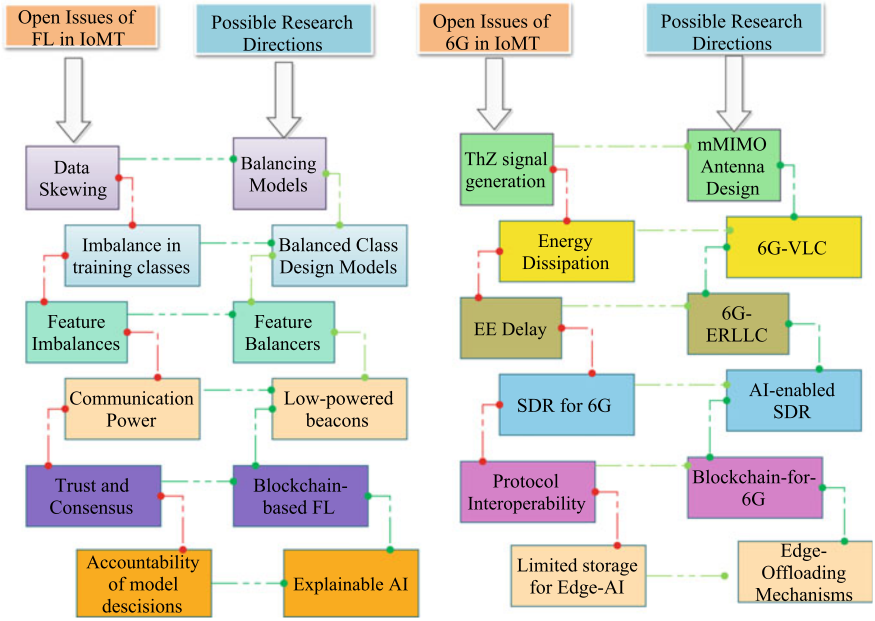

Open Issues of FL in IoMT
Possible Research Directions
Open Issues of 6G in IoMT
Possible Research Directions
Data Skewing
Balancing Models
ThZ signal generation
mMIMO Antenna Design
Imbalance in training classes
Balanced Class Design Models
Energy Dissipation
6G-VLC
Feature Imbalances
Feature Balancers
EE Delay
6G-ERLLC
Communication Power
Low-powered beacons
SDR for 6G
AI-enabled SDR
Trust and Consensus
Blockchain-based FL
Protocol Interoperability
Blockchain-for-6G
Accountability of model descisions
Explainable AI
Limited storage for Edge-AI
Edge-Offloading Mechanisms

Fig. 4 Open issues and possible research directions of integration of 6G and FL in IoMT ecosystems

models, thus, are required to communicate over small-sized beacon messages with low storage costs. Only necessary control bits are required during communication.

3. *Trust and Consensus*—Sensor devices are attack prone and maybe malignant in operations. Thus, trust is an important issue in decentralized model training. Thus, future research would shift towards effective model designs that incorporate blockchain and FL in achieving consensus once the model shares the local updates to central server.
4. *Accountability of model decisions*—Due to heterogeneous training on local data, there is often a debate on the accountability and interpretation of model decisions. Due to this, the central model decisions are often not accountable. Recently, explainable AI (XAI)-based solutions have been proposed, and XAI can be adopted in health care to include self-explainable and reasoned decision systems. The evaluation of model parameters and categorization of classes with proven reasoning would be the future guiding research for IoMT ecosystems.

5.2 6G for IoMT and Potential Research Directions

Next, we present the design challenges of 6G adoption in IoMT platforms. The details are presented as follows.

1. *ThZ signal generation*—6G would operate at THz signal bands, with support of optical communication at the core network. In the simulation, the generation of THz pulses is relatively straightforward, but in real-deployments, the signal generation is dependent on the fabrication of antenna and transmitter of appropriate size that can address service requirement in dense networks. In such cases, the study of optimized antenna design considerations to support massive users is a possible future area of research.
2. *Energy Dissipation*—Another key issue with the generation of THz bands is the energy dissipation of signals over a distance. 6G networks would exhibit higher losses due to higher signal spreading. A possible solution is to look towards visible light communication (VLC) as an alternative instead of THz signal generators. The extension of VLC to large areas would increase the bit error rates (BER) significantly, and thus low-optical BER error rate control mechanisms are required to be designed for 6G-VLC [19].
3. *End-to-end delay and 6G radio channels*—At physical layers, 6G communication testbeds are supported via software-defined radio (SDR) in the sub-THz range and are customized to provide network services at 2 GHz instantaneous bandwidth, with operation at 110–170 GHz. However, the channel requires AI-enabled SDR to improve the coding rate and reduce the neighbouring interference with large data sets.
4. *Protocol interoperability*—6G channels provide services-on-the-fly through virtualized networking functions. In many cases, the solutions are proprietary, and thus due to lack of standards in terms of message exchange formats and communication devices, the protocol conversions become difficult. Thus, heterogeneous services can operate uniformly if exchange formats are transformed into transactional blockchain ledgers, and thus blockchain for protocol interoperability is an emerging research domain.
5. *Limited Storage for Edge-AI*—6G-enabled IoMT nodes generate high traffic, and thus the achievable low latency of <1 ms to address quality-of-service degrades significantly. Thus, to reduce latency, edge devices run AI models to optimize traffic patterns and improve efficiency. However, over time, the edge-AI models tend to become bulky, which affects resource consumption. As IoMT networks, resources are critical. Thus, intelligent offloading mechanisms are required to be designed to manage the storage and computing requirements of AI models executing at edge nodes.

6 Conclusion

With the rise in privacy concerns over medical data, FL-based solutions are increasingly deployed in IoMT ecosystems. Recently, 6G networks have been envisioned to operate at THz bands at extremely low latency. The fusion of 6G and FL in IoMT addresses the latency constraints and provides intelligent control functions for the seamless management of nodes. The survey presents the reference architecture of 6G-assisted FL IoMT architecture. To ensure privacy against attacks, an (ϵ, δ) DP preserving FL model is presented, and layered architecture is discussed. However, the frameworks and protocols are still in native design phases, and thus, hence the survey discusses the challenges and emerging directions of the deployment of FL and 6G in IoMT ecosystems. The emerging research trends are highlighted, and suggestions for improvements are presented. Thus, the survey intends researchers to build a compelling fusion of 6G and FL schemes for IoMT that assures privacy-preserved business models with seamless user experience.

References

1. Bhattacharya P, Mehta P, Tanwar S, Obaidat MS, Hsiao KF (2020) Heal: a blockchain-envisioned signcryption scheme for healthcare IoT ecosystems. In: 2020 international conference on communications, computing, cybersecurity, and informatics (CCCI), Sharjah, United Arab Emirates, pp 1–6. https://doi.org/10.1109/CCCI49893.2020.9256705
2. MS Windows NT kernel description. https://www.rbccm.com/en/gib/healthcare/episode/the_healthcare_data_explosion. Accessed 28 Oct 2021
3. Verma A, Bhattacharya P, Bodkhe U, Ladha A, Tanwar S (2020) DAMS: dynamic association for view materialization based on rule mining scheme. In: The international conference on recent innovations in computing, Jammu, India, pp 529–544. Springer
4. Verma A, Bhattacharya P, Zuhair M, Tanwar S, Kumar N (2021) Vacochain: blockchain-based 5G-assisted UAV vaccine distribution scheme for future pandemics. IEEE J Biomed Health Inform
5. Kaiser MS, Zenia N, Tabassum F, Mamun SA, Rahman MA, Islam MS, Mahmud M (2021) 6G access network for intelligent internet of healthcare things: opportunity, challenges, and research directions. In: Kaiser MS, Bandyopadhyay A, Mahmud M, Ray K (eds) Proceedings of international conference on trends in computational and cognitive engineering. Springer, Singapore, pp 317–328
6. Elayan H, Aloqaily M, Guizani M (2021) Deep federated learning for IoT-based decentralized healthcare systems. In: 2021 international wireless communications and mobile computing (IWCMC), Harbin, China, pp 105–109. https://doi.org/10.1109/IWCMC51323.2021.9498820
7. Xu J, Glicksberg BS, Su C, Walker P, Bian J, Wang F (2021) Federated learning for healthcare informatics. J Healthcare Inform Res 5(1):1–19
8. Brisimi TS, Chen R, Mela T, Olshevsky A, Paschalidis IC, Shi W (2018) Federated learning of predictive models from federated electronic health records. Int J Med Inform 112:59–67. https://doi.org/10.1016/j.ijmedinf.2018.01.007
9. Qayyum A, Ahmad K, Ahsan MA, Al-Fuqaha A, Qadir J (2021) Collaborative federated learning for healthcare: multi-modal covid-19 diagnosis at the edge
10. Abdul Rahman S, Tout H, Ould-Slimane H, Mourad A, Talhi C, Guizani M (2021) A survey on federated learning: the journey from centralized to distributed on-site learning and beyond. IEEE Internet Things J 8(7):5476–5497

11. Huang L, Shea AL, Qian H, Masurkar A, Deng H, Liu D (2019) Patient clustering improves efficiency of federated machine learning to predict mortality and hospital stay time using distributed electronic medical records. J Biomed Inform 99:103291
12. Jiang JC, Kantarci B, Oktug S, Soyata T (2020) Federated learning in smart city sensing: challenges and opportunities. Sensors 20(21):6230
13. Li T, Sahu AK, Talwalkar A, Smith V (2020) Federated learning: challenges, methods, and future directions. IEEE Sign Process Maga 37(3):50–60
14. Choudhury O, Gkoulalas-Divanis A, Salonidis T, Sylla I, Park Y, Hsu G, Das A (2019) Differential privacy-enabled federated learning for sensitive health data. arXiv preprint arXiv:1910.02578
15. Yuan B, Ge S, Xing W (2020) A federated learning framework for healthcare IoT devices. arXiv preprint arXiv:2005.05083
16. Yin X, Zhu Y, Hu J (2021) A comprehensive survey of privacy-preserving federated learning: a taxonomy, review, and future directions. ACM Comput Surv (CSUR) 54(6):1–36
17. Mothukuri V, Parizi RM, Pouriyeh S, Huang Y, Dehghantanha A, Srivastava G (2021) A survey on security and privacy of federated learning. Future Gener Comput Syst 115:619–640
18. Reddy Gadekallu T, Pham QV, Bhattacharya S, Reddy Maddikunta PK, Liyanage M et al (2021) Federated learning for big data: a survey on opportunities, applications, and future directions. arXiv: 2110.04160 [cs.LG]
19. Singh A, Singh R, Bhattacharya P, Pathak VK, Tiwari AK (2020) Modern optical data centers: design challenges and issues. In: Giri VK, Verma NK, Patel RK, Singh VP (eds) Computing algorithms with applications in engineering. Springer, Singapore, pp 37–50

Rail Track Monitoring System Using Quantum Key Distribution in IoT Scenario

Nitya Chandra and W. Wilfred Godfrey

Abstract Quantum key distribution (QKD) is a secure communication method implementing a cryptography protocol involving the theory of quantum mechanics. It allows two entities to generate a shared random secret key known only to them. This secret key can then be used to encrypt and decrypt messages on the respective ends of the communication. Many researches have been conducted on different aspects of the same and researchers came up with the celebrated BB84 protocol. The protocol presents a way of exchanging the secret keys using the polarizations of photons. This paper introduces an application of QKD for a railway track monitoring system backed up with IoT sensors. An approach toward simulation of quantum key distribution between an IoT device and a server to encrypt the data sent to the server is discussed in the paper. It also demonstrates the simplicity of this method and its efficiency in producing a QKD simulation. Also, it is demonstrated that for the IoT security, the quantum key length thus generated is used.

Keywords QKD · IoT · Photon · Photon polarization · Network security

1 Introduction

As the world is moving toward the era of ever growing technology, witnessing the emergence of extremely high computationally powered systems is of no surprise. This power can achieve billions of calculations in just a matter of milliseconds. This can even lead to breaking of one of the most powerful encryption algorithms like the RSA, Deffie Hellman key exchange algorithm and others as well. One of the rescuing solutions could be the use of quantum key exchange. Since it is based on the principles of Quantum Physics and not on mathematical complexity, quantum key distribution (QKD) [1] is one of most the secure communication method which implements

N. Chandra (✉) · W. W. Godfrey
Information Technology, ABV-IIITM, Morena Link Road, Gwalior 474015, Madhya Pradesh, India
e-mail: nityachandra6@gmail.com

W. W. Godfrey
e-mail: godfrey@iiitm.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*, Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_16

a cryptographic protocol involving theory of quantum mechanics. It enables two parties to produce a shared random secret key known only to them, which can then be used to encrypt and decrypt messages [2]. A distinguishing property of quantum key distribution is the ability of the two participating users to detect the presence of an eavesdropper trying to gain access to the key or the data being transferred. This results from a fundamental property of quantum mechanics: the process of measuring a quantum system disturbs the system in one way of the other. A third party trying to intercept the communication must in some way measure it, and hence introducing detectable traces in the key. Quantum key distribution's use is limited to produce and distribute a key through a quantum link, not to carry any message data. This key can then be used with any chosen encryption algorithm to encrypt (and decrypt) a message, which can then be transmitted over a standard communication channel [3].

2 Literature Survey

2.1 Potential Applications of QKD

The quantum key distribution has huge number of applications in almost all domains. With the continuous development in the area of quantum computing, there is a need for uncompromising security in confidential data. Quantum key distribution protocols are proven secure if all devices are perfect (in terms of technologies and proper protocol operations). The major challenges in quantum communication are distance, secret key rate, size and cost of QKD devices. Quantum computing could have the most severe impact on the security of a nation. A large-scale quantum computer capable of deploying the Shor's algorithm on current encryption would have a devastating impact on virtually all Internet security [4].

Assuming all technical challenges could overcome, the performance of a QKD facility is at least as good as the classical network equivalent, e.g. in terms of key distribution rates and reliability, we discuss some of the potential applications of QKD systems [5].

- Anti-Skimming in ATM Transactions—A skimming attack occurs when the adversary attaches some equipment to detect or record the electronic data from the cards used in. These sensitive data can then be used for forgery purposes. If we install a QKD system in the banking sector, eavesdropping of the quantum key would be detectable resulting in aborting the transaction. Using a QKD quantum key in a one time pad ensures that PIN cannot be recovered by intercepted messages.
- Authentication in a Corporate Environment—It refers to the process of authenticating a person/entity/computer of what they claim to be. Corporate environments need the authentication to be fairly strict. Any unauthorized person should not be allowed in the buildings, which could happen provided a strong authentication system is not present. QKD can be used here for authentication and providing access control across the organization. Quantum key at different access points would ensure only authorized personal for such access.

- Electronic Voting—There are numerous areas in an election which could be improved like improving efficiency in registration and counting processes, reducing electoral fraud, etc. QKD has been already applied in this arena once in 2007 for voting processes in Geneva. The ballot information was encrypted using QKD keys, and then through the fiber optic link sent to the government data center for further processes.
- QKD with IoT systems—IoT combined with QKD opens up a playground to try for secure systems. For instance, health conditions of people such as ECG, blood pressure or information related to whereabouts of the beloved ones could be tracked through IoT sensors, and then the data can be relayed to some smart device like a smartphone. This sensitive data can then be sent to secure databases for further examination [6]. Another scenario could be the one demonstrated by this paper, rail track monitoring system using QKD. We have discussed about it closely in the later sections.

2.2 Quantum Key Distribution in Practice

The main security goals for the any communication are confidentiality, integrity and authentication. Same goes with the IoT devices. There are currently four companies offering quantum key distribution systems commercially [7], these include ID Quantique (Geneva), Magiq Technologies Inc. (NewYork), sequrenet (Paris) and Quintessencel labs (Australia). Practical implications of QKD include several forms of networks. These include but are not limited to:

- SECOQC—The Secure Communication based on Quantum Cryptography (SECOQC) network is the world's first computer network protected by quantum key distribution. This was implemented in October 2008, at a scientific conference held in Vienna. This network used 200km of standard fiber optic cable to interconnect six locations across Vienna and the town of St Poelten located in the west of Vienna.
- DARPA—The Defense Advanced Project Research Agency (DARPA) quantum network was developed by Harvard university, Boston university BBN technologies and Qnietiq. This is a 10-node quantum key distribution network, which has been running since 2004 in Massachusetts, USA.
- Swiss quantum—The main goal of the swiss quantum network project installed in the Geneva metropolitan was to validate the reliability and robustness of QKD in continuous operation over a long time period in a field environment.This has successfully completed the longest running project for testing QKD in field environment.

3 Methodology

3.1 Architecture

This section demonstrates an IoT scenario whose security is assured through the quantum key distribution technology [2].

3.1.1 Central Server

This a central point of the system from where all the decisions regarding any changes or measures to ensure safety and security of the tracks take place. All the real-time data would be relayed to this central server for processing from the controllers. It would contain all the databases and required setup for experimentation and calculation.

3.1.2 Controllers

These are the IoT devices supporting the QKD infrastructure. Each of the controller would be connected to the central server through a p2p (point-to-point) network which can be realized as an optical fiber connection. This would help in secret quantum key generation process. These controllers receive real-time track parameters from the IoT sensors and relay it to the central server.

3.1.3 IoT Sensors

These are the real-time sensors that are placed along the tracks, certain distance apart. This distance depends on what RF channel we use for communication between the controllers and the sensors. They sense the real-time track parameters like tilt, depth, temperature, etc., and send the fluctuations to the IoT controllers. These are connected to the IoT controllers over a Zigbee channel or a suitable RF channel.

3.2 System

The scene demonstrates a railway track monitoring system. IoT sensors would be used to monitor railway track parameters like tilt, temperature, etc., in real time and send the corresponding data to a controller. This could be different types of sensors like the Tilt sensor for measuring tilt, proximity sensor for detecting depth, etc. The data measured would then be relayed to a central server for analyzing/processing the data which could then be used to take preventive measures for the safety and security of the railway tracks. Here, for the QKD link (for sharing secret keys), optical fiber

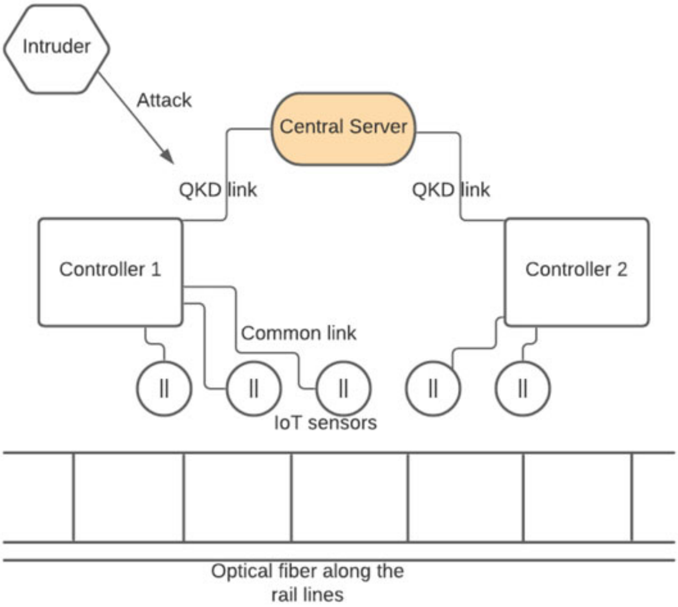


Fig. 1 Network topology for rail track monitoring system

cables can be laid along with the tracks or the digital communication link already being used for data communication could be used for secure and fast connection. Then the encrypted data can be shared between the controller and IoT sensors.

Figure 1 represents the network topology of the system, containing 2 IoT controllers, 5 IoT sensors and the central server. A potential attacker might want to eavesdrop the sensitive data or tamper it. This may lead to harmful effects on life and property. Hence must be taken care of. So, we also demonstrate an attack on the system which by the aid of QKD would get detected, since QKD works on the laws of physics. Detecting an attack on the system leads to the termination of the generation of keys and decrements the key length.

3.3 Application Mechanism

The topology was simulated on ns3 [6] network simulator using the algorithms described below. The base of the designed protocol is BB84 [4] underneath. The controller produces the first step and generates photon stream after polarization. The polarization of each photon is chosen at random. It transmits these polarized photons one at a time through the quantum link and takes a note of the uncertainty of polarization chain such that there is no way of guessing polarization any of the photons by the receiver. The receiver has two filters, a rectilinear and a diagonal filter. When the photons reach the destination, the receiver directs them one of the filters at random and takes a note of what photon went to which filter, this is the basis detection. The receiver then transmits its basis through the classical channel. The points/indices where both their basis comes out to be equal, they consider it a part of the key. In case of an attack by an adversary, it's the polarization of a photon which gets affected.

At the beginning, declaration of the server and the controller array takes place having its elements as random numbers between 1 and 4 each representing polarization of the photon. The values 1, 2, 3 and 4 correspond to 0, 45, 90 and 135 degree polarization of photon, respectively. On comparing each of the random values in the arrays, if there is a match, we assign it to the key array and increment the key length by 1. We also keep a track of the number of photons having different polarizations. *N* represents the number of quantum bits (qbits) or the number of photons we are using for the process.

Algorithm 1 Generate secret keys

```
cont[] = 0          \\initializing controller array
server[] = 0        \\initializing server array
key[] = 0           \\initializing key array
len = 0             \\length of key
for i:=1 to N do:
    cont[i] = rand(1, 4)
for i:=1 to N do:
    server[i] = rand(1, 4)
for i:=1 to N do:
    if: cont[i] equals server[i] then
        key[i] = cont[i]
        len += 1
    if: key[i] equals 1 then
        pol_0 += pol_0
    if: key[i] equals 2 then
        pol_45 += pol_45
    if: key[i] equals 3 then
        pol_90 += pol_90
    if: key[i] equals 4 then
        pol_135 += pol_135
```

Above is represented a way of generating and coming up with a secret key for the QKD process.

3.4 Attack Detection

This section proposes the algorithm for the attack mechanism. This could be a potential man in the middle attack to gather or tamper the information in transmission.

It is assumed that each IoT controller has an attacker to simulate the ability for the attacker to detect a certain length of the key. The algorithms proceeds as follows:

Algorithm 2 Attack mechanism and detected key length

```
attacker[] = 0     \\initializing attacker array
detectedLen = 0    \\detected key length by attacker
p=1
for: i:=1 to N
    attacker[i] = p
    p += 1
    if: p equals 4 then
        p = 1
    if: attacker[i] equals cont[i] &&
                server[i] equals attacker[i] then
        detectedLen = detectedLen + 1
```

The attacker array is declared having the same length to the other two, and after assigning random values to it as well we check if all the three (controller, server and the attacker) values are equal at what indices, those values contribute to the final value of the detected key length.

4 Results

The use case is a railway track monitoring system which uses QKD technology for secure transaction of track's real-time data to a central processing station. Several IoT sensors connected along the railway lines monitor track parameters and relay it to the controllers. And then to the control server for analysis. The quantifying metric is the key length (bits) of the key thus generated. Further, we demonstrated a man in the middle attack which results in reduced key generation and hence the key length.

A 1000 cell array was used, representing the number of photons required in the key generation process. Two QKD-enabled controllers and a central server. The average values of the key length and each polarization counts for 5 iterations are shown below for both the controllers.

Figure 2 represents the resultant key length generated for controller 1, and the length of generated key is 259. The photon polarization counts are as follows: 68 photons with 0 degree, 60 photons with 45 degree, 71 photons with 90 degree and 60 photons with 135 degree.

Figure 3 represents the resultant key length generated for controller 2, and the length of generated key is 236. The photon polarization counts are as follows: 60 photons with 0 degree, 70 photons with 45 degree, 56 photons with 90 degree and 50 photons with 135 degree.

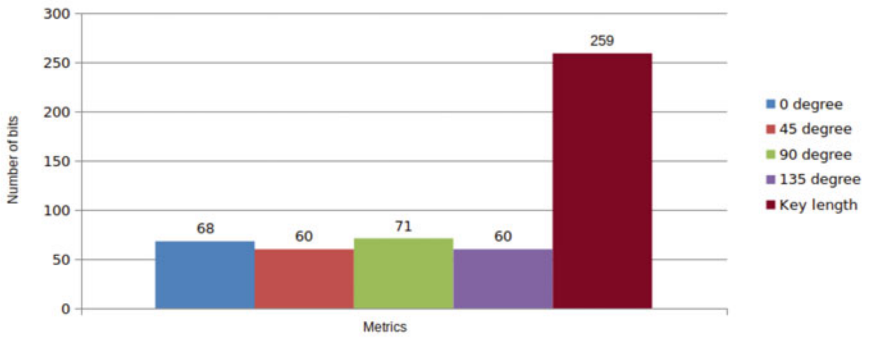


Fig. 2 Polarization of photons and length of key for controller 1

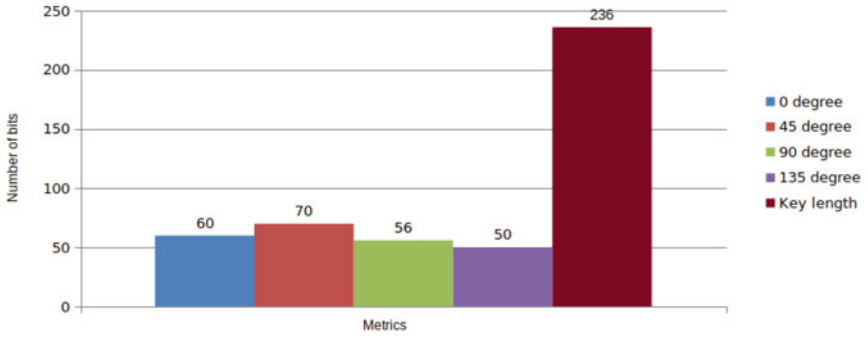


Fig. 3 Polarization of photons and length of key for controller 2

Then, four cases were considered each with 100, 200, 300 and 400 number of photons or initial quantum bits (qbits). After 5 iterations on each of the cases, the average key length agreed by the parties was obtained. Figure 4 represents the results as we got the average agreed key lengths as 29, 48, 79 and 104, respectively.

On these average agreed key lengths was simulated the attack (again 5 times for each case) on the system as described in algorithm 2. Figure 5 shows the simulation results as the average number of bits tampered by the attacker or the detected key length by the attacker for each of the 4 cases above. We can conclude that the final key length after the attack remains only 76 as for case 1 with 104 bits of agreed length. Similarly, we can calculate for other cases as well.

The above diagrams represent the key length of the secret key established among the two parties during QKD. From the results, we can say that after an attack, the length of the secret key decreases, this is because the parties can detect the attack and reject the portion which is tampered. If further, the attacker continues with the tampering, the key length keeps on decreasing and finally, it would become 0. This would result in communication being suspended.

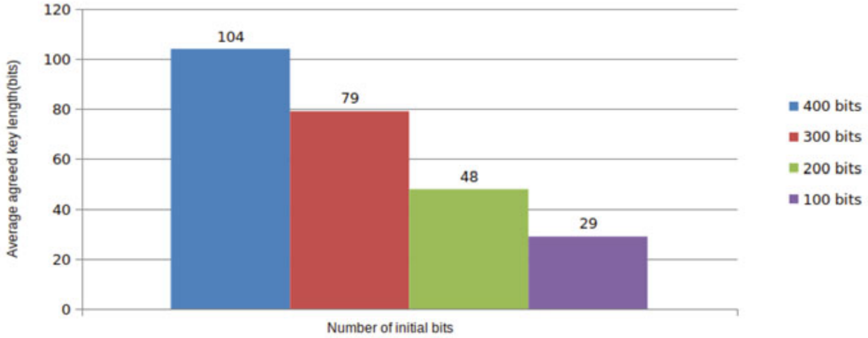


Fig. 4 Average agreed key lengths among the parties

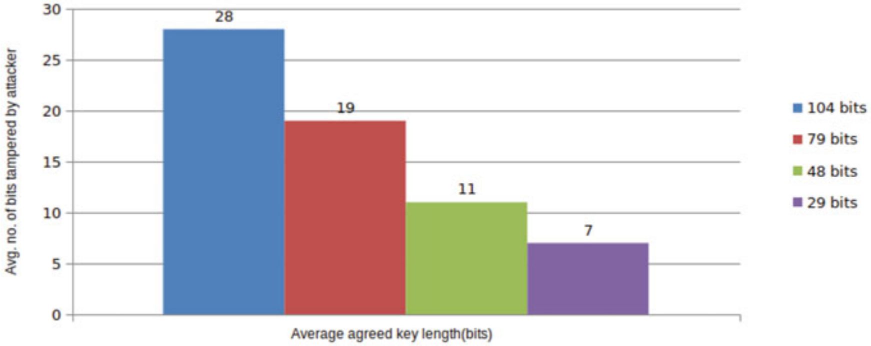


Fig. 5 Number of bits detected by the attacker for each case

5 Conclusion

In this paper, a method for simulating [8] quantum key distribution between IoT controllers and the central server is presented and that too in a secure way. Further, we saw generating and measuring the resultant key length for the pre-decided number of initial photons. This key can then be used for symmetrical encryption for ensuring data protection for IoT devices. Moreover, we saw the performance of the system under the influence of an intruder in the simulations. The length of the final key is enough to be used in symmetric cryptographic algorithms like one time pad, AES/DES by the IoT devices even in the presence of an attacker. If the attack continues, the communication may get suspended. Thus, there is a possibility of Denial of Services attack on the system.

There are some other challenges [9] as well that can be addressed for the future work. The current system depends on the authenticity of the central server, if that is compromised at any point, the rail monitoring system would collapse. Another possibility would be of the controllers getting hacked. They could relay malicious data to the central server which could even have fatal results, so there needs to be viable check. Future systems can think of a more robust security which could handle the topology firmly enough to fight not only current attack which makes the system vulnerable but also any upcoming new attacking techniques that an eavesdropper can think of.

References

1. Kowsalya T, Sukirtha S, Krithika S. Quantum key distribution for internet of things (IoT)—a review
2. Pljonkin A, Rumyantsev K, Singh PK (2017) Synchronization in quantum key distribution systems. Cryptography 1(3). https://doi.org/10.3390/cryptography1030018
3. Zhang H, Ji Z, Wang H, Wu W (2019) Survey on quantum information security. China Commun 16(10):1–36
4. Parker E (2021) Commercial and military applications and timelines for quantum technology. RAND Corporation, Santa Monica, CA. https://doi.org/10.7249/RRA1482-4
5. Cobourne S et al (2011) Quantum key distribution protocols and applications. Surrey TW20 0EX, England
6. Routray SK, Jha MK, Sharma L, Nyamangoudar R, Javali A, Sarkar S (2017) Quantum cryptography for IoT: aperspective. In: 2017 international conference on IoT and application (ICIOT). IEEE, pp 1–4
7. Pljonkin A, Singh PK (2018) The review of the commercial quantum key distribution system, pp 795–799. https://doi.org/10.1109/PDGC.2018.8745822
8. Al-Mohammed HA, Yaacoub E (2022) New way to generating and simulation QKD. In: Proceedings of sixth international congress on information and communication technology. Springer, pp 801–809
9. Using quantum key distribution for cryptographic purposes: a survey. Theor Comput Sci 560:62–81 (2014); Theoretical aspects of quantum cryptography celebrating 30 years of BB84. https://doi.org/10.1016/j.tcs.2014.09.018
10. Abd El-Latif AA, Abd-El-Atty B, Venegas-Andraca SE, Elwahsh H, Piran MJ, Bashir AK, Song O-Y, Mazurczyk W (2020) Providing end-to-end security using quantum walks in IoT networks. IEEE Access 8:92687–92696
11. Mehic M, Maurhart O, Rass S, Voznak M (2017) Implementation of quantum key distribution network simulation module in the network simulator NS-3. Quantum Inf Process 16(10):1–23
12. Shor PW, Preskill J (2000) Simple proof of security of the bb84 quantum key distribution protocol. Phys Rev Lett 85(2):441

Evaluating Tip Selection Algorithms for IOTA Blockchain

Shreya Purohit and Tushar Champaneria

Abstract IOTA blockchain is an alternative to blockchain, which is utilized for millions of devices connected to an Internet of Things network. These devices can communicate data with other devices in the form of transactions using IOTA. The IOTA tangle is the network of such devices. To complete a new transaction, tips of the tangle must be discovered, which is accomplished through the use of several tip selection methods. The process of picking two tips in the IOTA tangle to which the incoming transaction connects is known as tip selection. The study of various tip selection methods used in IOTA blockchain is described in this research, and tip selection strategies are evaluated with respect to time and varied number of node configurations using JGraphT library. Finally, a comparison of all tip selection algorithms is shown, with the adaptive random walk algorithm outperforming the others.

Keywords Blockchain · IOTA · Tip selection algorithm

1 Introduction

IOTA is a decentralized database that will store and execute transactions between billions of Internet of Things devices and machines. IOTA has a few features that are causing individuals to switch from blockchain to IOTA. These advantages are listed below:

- Zero Transaction fees: There are no fees for conducting a transaction because each node in a tangle contributes by confirming two nodes.

S. Purohit (✉) · T. Champaneria
Computer Engineering Department, L.D. College of Engineering,
Ahmedabad 380015, Gujarat, India
e-mail: 2018csshe015@ldce.ac.in

T. Champaneria
e-mail: tac@ldce.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_17

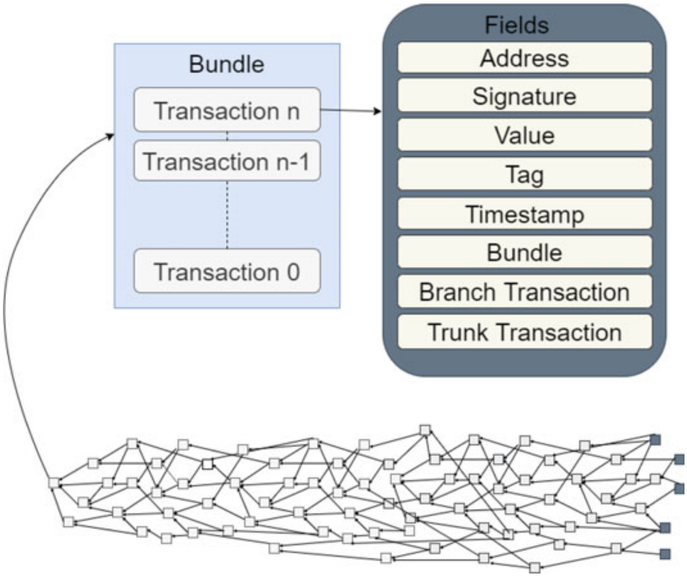


Fig. 1 Basic diagram of tangle with bundle

- Scalability: The transaction data do not need to be saved in each node in a tangle when it is attached to it.
- Data Integrity: The signature of each transaction is validated before it is executed. Additionally, each time a new key pair must be computed.
- Fast configuration: The transactions are executed in parallel due to the DAG data structure.
- Micro-transaction: The client can afford micropayments because IOTA has no transaction fees [1].

The most important factor is IOTA's data structure, which is a directed acyclic graph (DAG) that generates a tangle structure [2, 3]. As a result, nodes in a DAG are connected in a specific direction without forming a cycle, i.e., nodes cannot refer back to themselves.

When the client generates a new transaction, it will be represented in the form of a bundle, as previously stated. It must be attached to a tangle after it is created. This is accomplished by connecting the bundle to the tangle's preceding two nodes. These two tips will ensure that your incoming transaction is approved. These two tips are for transactions that do not have parents. Instead of picking two tips at random (or always using the most recent two), the tips are chosen using a tip selection algorithm. Unweighted random walk algorithm, weighted random walk method, greedy weighted random walk algorithm, and adaptive random walk algorithm are some of the tip selection strategies available [4–6].

Figure 1 illustrates the significance of a bundle, which is a data structure that contains one or more transactions where the transaction is an operation which can be of type input or output. To send a transaction to a node (full), you need to bundle them. These transactions have some fields which will store the information about a particular transaction in a bundle. From which, one of them is a bundle that consists of bundle hash. Another is address which is self-explanatory, i.e., store address of the transaction. The value field contains the token that we want to transfer. The field signature consists signature of the transaction, which is calculated by using the Winternitz one-time signature scheme (W-OTS). Branch transaction and trunk transaction filed consists of the pointers where trunk will chain all transactions in bundle and trunk of last transaction points to tip 0.

The tangle structure can be generated or formed with the help of a Java library named JGraphT [15]. This library offers classes and functions which will help to create directed acyclic graph. If we can simulate the behavior of tangle, we can easily evaluate walking algorithms without falling into the complexity of the IOTA tangle itself.

In Sect. 1, various tip selection algorithms will be discussed; then, in Sect. 2, implementation of those algorithms is discussed, which is performed using library JGraphT, and in Sect. 3, the result observed on the basis of experiments which is performed by using a different number of nodes will be shown along with convergence graph. Finally, the summarization of this paper will be done in Sect. 4.

2 Tip Selection Algorithms

Tips are the unconfirmed node of a tangle. And in a tangle, tips are the leaf nodes of DAG. Now, whenever a new transaction takes place, it will get connect to the two tips to become a part of the tangle. For getting these tips, random walk algorithms are applied on the DAG, which will result in two tips [4]. There are various walking algorithms that will place two walkers on the genesis (the starting node of tangle), which will follow different strategies and results in two tips. The selection of a good algorithm results in healthy growth of tangle, i.e., in stable and secure way.

2.1 Unweighted Random Walk

An unweighted random walk is a basic and simple walking algorithm in which the walkers will choose the next node to be hop randomly. However, this algorithm is suitable for networks due to some weaknesses. This algorithm will allow the selection of lazy nodes and malicious nodes where lazy nodes are the tips that will get attached to the older transactions in a tangle by confirming them as these nodes are too lazy to do a traverse and confirm the nodes to get attached to a newer node in a tangle, while malicious nodes will increase the number of tips by issuing many transactions that approve a fixed pair of transactions.

Algorithm 1: Unweighted random walk algorithm

```
Result: two tips
rw=new RandomWalkIterator(dag, start_ver);
while rw.hasNext() do
    walk = walk + rw.next();
end
return walk;
```

The time complexity of Algorithm 1 is O(log n).

The function named RandomWalkIterator(dag, start_ver) has two parameters, dag, and start_ver where dag is the tangle on which traversal is done and start_ver is the starting vertex from where traversal takes place. This function will start the traversal from the start_ver till the dag has the next vertices.

2.2 Weighted Random Walk

This algorithm will overcome the problems experienced in unweighted random walk algorithm. The basic idea of this algorithm is that, at every time when selection of a node is done by walker to move further, the probability of selecting a node is calculated on the basis of cumulative weight. The cumulative weight of the node is the sum of the weight of that node and all the nodes that confirm that node directly or indirectly, which is calculated using Eq. (1), where the meaning of $y \Rightarrow x$ is, y is a set of those nodes that are confirming node x directly or indirectly.

$$\text{Cumulative\_Weight}_x = \text{Weight}_x + \sum_{y \Rightarrow x} \text{Weight}_y \tag{1}$$

Based on Eq. (2) given, we can find the probability of moving a walker to which node. Consider, right now, our walker is on node x, and it has a choice to move on three nodes that are confirming node x directly. Let those nodes be p, q, r, and then, the walker will calculate the probability to select the next node to hop. Whichever node has a higher probability, the walker will move on to that node.

$$P_x y = exp(-\alpha(W_x - W_y))(\sum_{z \Rightarrow x} exp(-\alpha(W_x - W_z)))^{-1} \tag{2}$$

Here, W stands for cumulative weight, where α is a positive integer, and the meaning of $y \Rightarrow x$ is a set of those nodes that are confirming node x directly or indirectly.

In [16], authors discussed about Markov Chain Monte Carlo methodology, or MCMC, which is a method of establishing a rule for determining the probability of each step in a random walk. In a Markov chain, each step does not depend on the one before it but instead follows a predetermined rule. If we set α to zero, the weighted random walk algorithm will behave as an unweighted random walk. Furthermore, if we set α to very high, it will behave like a superweighted random walk algorithm. If we have a strike the value of α in between these low and high ends, we can find a decent balance between punishing laziness and not leaving too many tips behind. As the value of α is selected in a random manner, there are many good tips the algorithm is leaving behind. So, the value of $\alpha\alpha$ needs to be chosen in an adaptive manner. That is where the need of adaptive random walk arises.

Algorithm 2: Weighted random walk algorithm

```
Result: two tips
rw=new RandomWalkIterator(dag, start_ver);
while rw.hasNext() do
    temp = rw.next();
    alpha = random.Next(1); ListOfEdges.add(dag.outgoingEdgesof(temp));
    for $i = 0, i <= ListOfEdges.size, i++$ do
        cumulativeWeightXtoY=cumulativeWeight(dag, temp) -
        cumulativeWeight(dag, list_ed_s.get(i));
        cumulativeWeightXtoZ=(cumulativeWeight(dag,in_ver.get(0)))-
        (cumulativeWeight(dag,in_ver.get(i)));

        probability = exp((-1) * cumulativeWeightXtoY)*(sum(pow((exp((-1)*
        cumulativeWeightXtoZ)),(-1))));
        if probability >= min then
            max = probability;
            walk1 = ListOfEdges.get(i);
        end
    end
    walk = walk + walk1;
end
return walk;
```

The time complexity of Algorithm 2 is $O(xylog(n))$.

2.3 Adaptive Random Walk

The main intention of this algorithm is to overcome the shortcomings and gain the benefits of the previous algorithm. The essence of this algorithm is α used in the Eq. (3) [6]. Variable α can adapt itself on the basis of network situation. If this α is 0, then the algorithm will behave as unweighted random walk algorithm; the reason is cumulative weight term is multiplied to α. With an increase in variable α, the cumulative weight is increased. When α is too high, then the algorithm will behave as a superweighted random walk algorithm.

$$\alpha_x = \frac{6}{n\sqrt{2}} \times \text{STD}\{W_z | z \in DC_x\} \quad (3)$$

Here, n is the cumulative weight of genesis. DC_x is a number of nodes directly confirming node x.

Algorithm 3: Adaptive random walk algorithm

```
Result: two tips
rw=new RandomWalkIterator(dag, start_ver);
while rw.hasNext() do
    temp = rw.next();
    ListOfEdges.add(dag.outgoingEdgesof(temp));
    alpha = (6/(cumulativeWeight(dag,start_ver) * sqrt(2))) * stddev(temp,dag);
    for $i = 0, i <= ListOfEdges.size, i++$ do
        cumulativeWeightXtoY=cumulativeWeight(dag, temp) -
        cumulativeWeight(dag, list_ed_s.get(i));
        cumulativeWeightXtoZ=(cumulativeWeight(dag,in_ver.get(0)))-
        (cumulativeWeight(dag,in_ver.get(i)));

        probability = exp((-1) * cumulativeWeightXtoY)*(sum(pow((exp((-1)*
        cumulativeWeightXtoZ)),(-1))));
        if probability >= max then
            max = probability;
            walk1 = ListOfEdges.get(i);
        end
    end
    walk = walk + walk1;
end
return walk;
```

The time complexity of Algorithm 3 is $O(xy\log(n))$.

2.4 Greedy Weighted Random Walk

We tested this algorithm with the same algorithm as for weighted random walk with a little difference. Consider, right now, our walker is on node x, and it has a choice to move on three nodes that are confirming node x directly. Let those nodes be p, q, r, and then, the walker will calculate the probability to select the next node to hop. Now, the difference is that whichever node having a lower probability the walker will move on to that node.

Algorithm 4: Greedy weighted random walk algorithm

```
Result: two tips
rw=new RandomWalkIterator(dag, start_ver);
while rw.hasNext() do
    temp = rw.next();
    ListOfEdges.add(dag.outgoingEdgesof(temp);
    for $i = 0, i <= ListOfEdges.size, i{+}{+}$ do
        if cumulativeWeight(dag, ListOfEdges) >= min then
            min = cumulativeWeight(dag, ListOfEdges);
            walk1 = ListOfEdges.get(i);
        end
    end
    walk = walk + walk1;
end
return walk;
```

The time complexity of Algorithm 4 is $O(mlog(n))$.

All the algorithms discussed above have a different way of approach where an algorithm will select the tip randomly without weight, while another one will select using weights. Also, they can be selected with the help of weight with a variable named alpha. After a discussion of algorithms, their time complexities are given. An unweighted random walk algorithm is a naive algorithm. This is the reason its time complexity is $O(logn)$.

IOTA has been extensively studied in the past using synthetic data which is shown in [7–13]. These works build their own applications and evaluate system performance using the transaction data generated in a simulated environment [14]. To evaluate different tip selection algorithms, an environment is created which performs various random walk algorithms with different number of nodes in the tangle.

3 Implementation

For the implementation of the algorithms specified above, we are using a library named JGraphT [15]. It is a free Java class library including graph-theory objects and algorithms. On a Linux operating system with 4GB of RAM and 6 processors, we used Java platform (needs JDK 1.8 or later starting with JGraphT 1.0. 0).

This library can be used for many purposes, among which one is used to creating tangle, which is nothing but a DAG data structure used by IOTA. This library provides different APIs which is suitable to implement the algorithms discussed in Sect. 2. Figure 2 illustrates the flow of implementation of DAG. For creating DAG, a class named DirectedAcyclicGraph⟨V, E⟩ is used, which will give tangle as an output. The input of this class is V and E, where V is a set of vertices and E is a set of edges. Here, V is vertices of type string, and E is edges of type default edge. These V and E are gener-

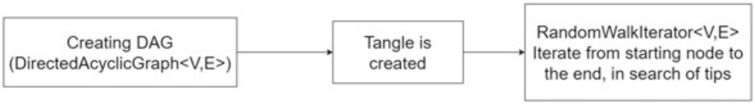

Creating DAG
(DirectedAcyclicGraph<V,E>)
Tangle is created
RandomWalkIterator<V,E>
Iterate from starting node to the end, in search of tips

Fig. 2 Flow of implementation

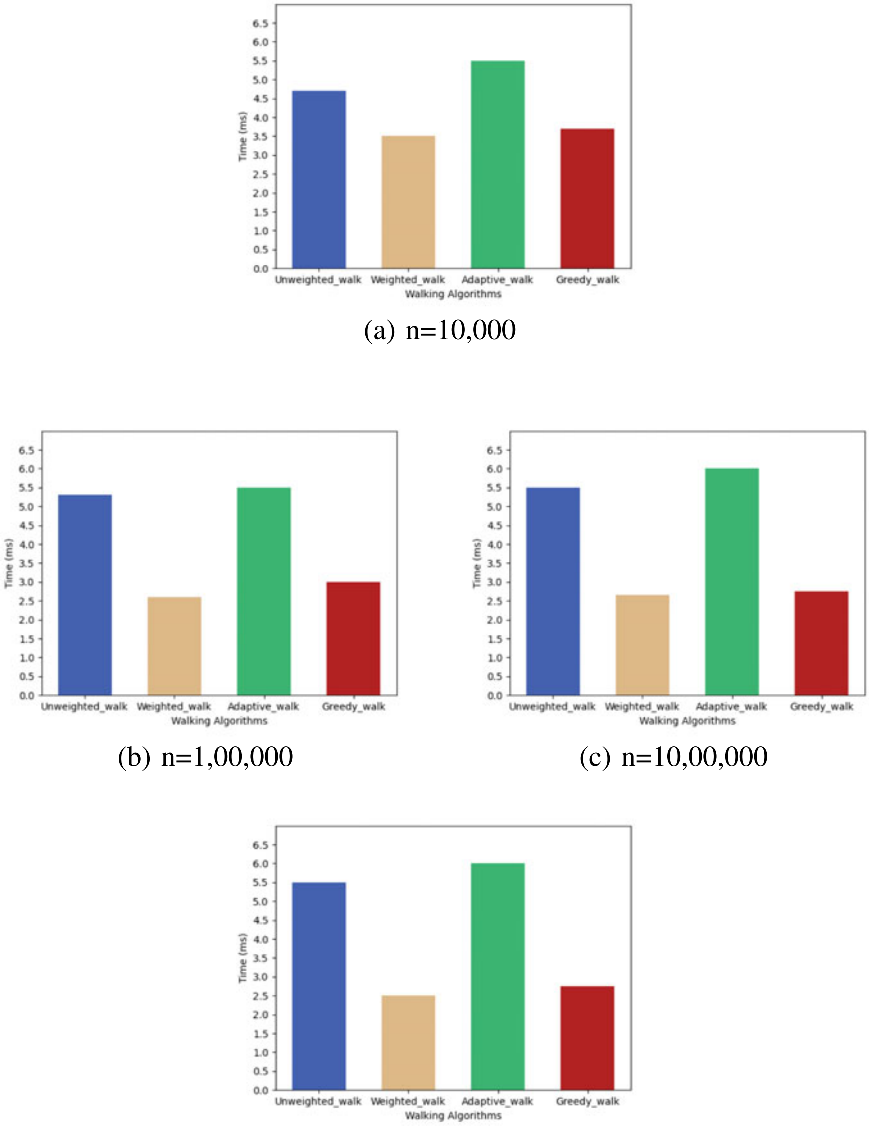
ated randomly and stored in CSV files. To create a tangle, vertices and edges are added to DAG with the help of custom function named as add_ver() and add_edge(). After tangle is generated, various tip selection algorithms are implemented in which two walkers are placed on genesis for selecting tips. These algorithms are created in a custom function named randomwalkiterator(), weighted randomwalkiterator(), greedy-weighted randomwalkiterator(), and adaptive randomwalkiterator(). In all these algorithms, walk iterator is placed on genesis for selecting the next node randomly, which is done by using a class named RandomWalkIterator⟨V, E⟩. For better understanding, let consider any algorithms discussed above, say, adaptive random walk algorithm. In this algorithm, DAG is generated using a class named DirectedAcyclicGraph⟨V, E⟩. The nodes or vertices and edges are added using custom function add_ver() and add_edge(). After DAG generation walk will be initiated by two walkers by using a class named RandomWalkIterator⟨V, E⟩. Alpha parameter is calculated using Eq. 3 which will call custom function std_dev for calculating standard deviation and also another custom function named CumulativeWeight() which uses outgoingEdgesOf() function of library used which helps in knowing all the incoming edges of a particular node directly or indirectly.

4 Result

By using the above implementation, different readings against the four tip selection algorithm (discussed in Sect. 2) are taken with a different number of nodes as an input. This number of nodes, say n, represents the number of transactions done so far to form a tangle. The result is represented graphically by specifying different algorithms discussed above on Y-axis and the time taken by them on X-axis (in ms). Figure 3a plotted the graph with 10,000 as the value of n, while Fig. 3b represented the comparison with $n = 1,00,000$, Fig. 3c, d plots the graph with $n = 10,00,000$ and $n = 20,00,000$, respectively.

On the basis of the graphs, which are plotted with a different number of nodes, a convergence graph is drawn shown in Fig. 4, which compares different walking algorithms with a different number of nodes. In the graph below, y-axis is labeled as n, which is a number of nodes. X-axis labeled as time in ms. These graphs are drawn by calculating the average of five readings which is taken for the same number of nodes in a particular algorithm.

Here, the random walk algorithm is taking more time if the number of nodes increases. No matter how simple unweighted random walk algorithm is, it is not



(a) n=10,000

(b) n=1,00,000

(c) n=10,00,000

(d) n=20,00,000

Fig. 3 Comparison between various tip selection algorithm with respect to no. of node

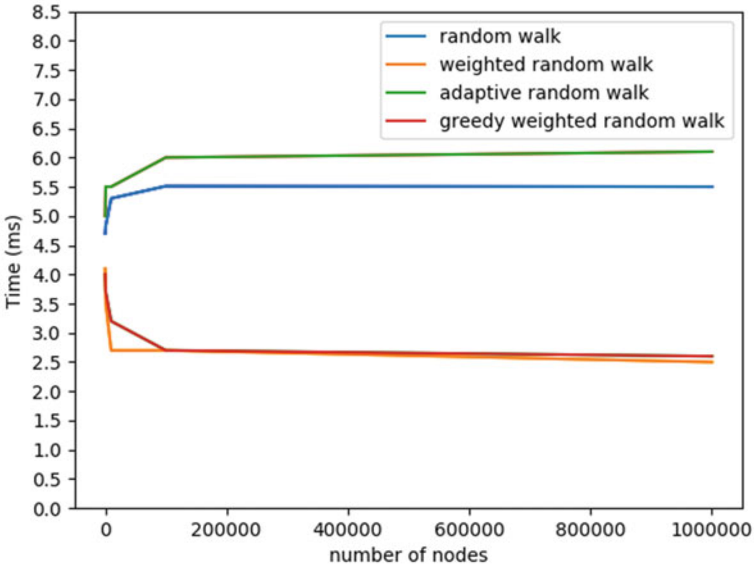


Fig. 4 Comparison between random walk algorithms for tip selection

suitable for the IOTA network as it can end up finding a lazy node or malicious node. So, it is advisable not to use this algorithm for any practical scenario. Another one is adaptive random walk, whose graph also increases with an increase in the number of nodes. It happens because the computation time of alpha takes place by every walker in the algorithm. However, this algorithm will not end the tip selection with a lazy node or malicious node. It will always select good nodes (the nodes rather than lazy and malicious nodes) by considering the network conditions. Here, network conditions are the condition which is seen from the sight of walker placed on a node in search of next node to hop in. Suppose this walker has many paths ahead from which one of the nodes has to be selected in order to reach the tip. By keeping this situation in priority, the alpha will be calculated, and it will be proportional to the standard deviation of cumulative weights of the set of the nodes that are encountered by a walker. At the same time, greedy and weighted random walk behaves in a similar fashion with a very minute difference. The reason behind their behavior is the algorithm, which is the same with a bit of difference, i.e., selection of nodes in greedy weighted random walk algorithm will be done on the basis of minimum cumulative weight, while weighted random walk algorithm will does on the basis of maximum cumulative weight. Both algorithms will prevent the network from potential risks like lazy nodes and malicious nodes. But, they do injustice to many good nodes by only selecting the nodes having maximum and minimum cumulative weight.

5 Conclusion

In this paper, simulation of IOTA tangle is done using the JGraphT library. One of the essential aspects of IOTA, i.e., tip selection is implemented. There are various algorithms such as the unweighted random walk algorithm, weighted random walk algorithm, adaptive random walk algorithm, and greedy weighted random walk algorithm for tip selection. So, by comparing these algorithms with a different number of nodes, a conclusion can be drawn that for a varying number of nodes, unweighted random walk, and adaptive random walk algorithm are taking more time than the Weighted random walk algorithm and greedy weighted random walk algorithms. Although adaptive random walk algorithm takes slightly more time (i.e., 1 ms) to execute but gives some trade-off, it will ensure selection of good nodes. In contrast, weighted random walk and greedy weighted random walk algorithms are faster but tend to ignore good nodes, thus increasing the number of iterations.

References

1. Hellani H, Sliman L, Samhat AS, Exposito E Computing resource allocation scheme for DAG-based IOTA nodes, 9 July 2021
2. Benčić FM, Žarko IP (2018) Distributed ledger technology: blockchain compared to directed acyclic graph. In: Proceedings of international conference distributed computer system, vol 2018-July, pp 1569-1570
3. Verzijl D, Dervojeda K, Jorn S-K-F, Nagtegaal F, Probst L, Frideres L (2019) Proceedings of the third international scientific conference ‘intelligent information technologies for industry’ (IITI’18), Univ. West. Aust., vol 875, p 17
4. Popov S (2018) IOTA whitepaper v1.4.3. New Yorker 81(8):1–28
5. C. Team and I. Foundation (2019) The Coordicide Team, IOTA Foundation, pp 1–30
6. Chafjiri FS, Mehdi Esnaashari Esfahani M (2019) An adaptive random walk algorithm for selecting tips in the tangle. In: 2019 5th international conference on web research ICWR 2019, pp 161–166
7. Kusmierz B (2017) The first glance at the simulation of the Tangle: discrete model. IOTA Found. WhitePaper, pp 1-10
8. Kusmierz B, Staupe P, Gal A (2018) Extracting tangle properties in continuous time via large-scale simulations. Technical report. Working paper. Accessed on 23 Aug 2018
9. Kusmierz B, Gal A (2018) Probability of being left behind and probability of becoming permanent tip in the tangle, vol 2
10. Bottone M, Raimondi F, Primiero G (2018) Multi-agent based simulations of block-free distributed ledgers. In: 2018 32nd international conference on advanced information networking and applications workshops (WAINA). IEEE, pp 585–590
11. Fan C, Khazaei H, Chen Y, Musilek P (2019) Towards a scalable DAG based distributed ledger for smart communities. In: 2019 IEEE 5th world forum on internet of things (WF-IoT). IEEE, pp 177–182
12. Popov S, Saa O, Finardi P (2019) Equilibria in the tangle. Comput Ind Eng 136:160–172
13. Gardner R, Reinecke P, Wolter K (2020) Performance of tip selection schemes in dag blockchains. In: Mathematical research for blockchain economy. Springer, pp 101–116
14. Guo F, Xiao X, Hecker A, Dustdar S (2020) Characterizing IOTA tangle with empirical data. GLOBECOM 2020—2020 IEEE global communications conference

15. JGraphT. Available https://github.com/jgrapht/jgrapht
16. Silvano WF, Marcelino R Iota tangle: a cryptocurrency to communicate internet of things data. Applied Research Laboratory, Federal University of Santa Catarina, Ararangu’a, Santa Catarina, Brazil, May 21, 2020

LEAPS: Load and Emission Performance Characteristics for Sensor-Driven Green Transport Systems

Pronaya Bhattacharya, Chandan Trivedi, Janmay Bhatt, Prem Desai, and Sudeep Tanwar

Abstract The rise of urban vehicular traffic, mostly in public transportation, has increased the amount of carbon and nitrous emissions, thereby affecting the environmental conditions. Thus, researchers have shifted toward the design of green transport systems (GTSs) that require sensor-driven ecosystems. In GTS, harmful emission sensors are installed on vehicles, to measure the emission characteristics. However, the emission has a direct relationship with load and GTS routes, which is not studied in detail. Owing to the research gap, in this paper, we propose a novel scheme, *LEAPS* that proposes a relationship between the offered load and emission in a GTS ecosystem. In *LEAPS*, we considered an intelligent route mechanism based on start and stop points, with varying loads at each point. To lower the emissions, we consider route optimization that selects fewer stop points between pre-decided destination points. The proposed results indicate the efficacy of the scheme against conventional approaches.

Keywords Green transport systems · Route optimization · Sensor-driven ecosystems · Toxic emissions characterstics

P. Bhattacharya · C. Trivedi (✉) · J. Bhatt · P. Desai · S. Tanwar
Institute of Technology, Nirma University, Ahemadabad, Gujarat, India
e-mail: chandan.trivedi@nirmauni.ac.in

P. Bhattacharya
e-mail: pronoya.bhattacharya@nirmauni.ac.in

J. Bhatt
e-mail: 19bce025@nirmauni.ac.in

P. Desai
e-mail: 18bce186@nirmauni.ac.in

S. Tanwar
e-mail: sudeep.tanwar@nirmauni.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_18

1 Introduction

Over the last decade, with the rise in carbon emissions and unregulated transportation systems, the search toward green transportation has gained prominence. In green transport systems (GTSs), we tend to reduce vehicular emissions, which leads to adverse effects on nearby communities. It is predicted in [1] that carbon dioxide (CO_2), which is a greenhouse gas and also an air pollutant, and carbon dioxide (CO_2) emissions would rise over 50% by the year 2030 to what it was in 2006. Hence, there is a stringent need to develop environment-friendly solutions or upgrade current transport ecosystems to make them environment-friendly.

Thus, researchers have shifted toward transport solutions that control harmful emissions. In such solutions, sensor-based components (emission sensors like CO_x and NO_x) measure the real-time emissions and send the reading to the servers for analysis. Thus, through Internet-of-Things (IoT) sensors, load and emission control can be performed that enables the vision of GTS.

Statistically, at global front, SO_2 and NO_2 and O_3 would increase in the near future. Thus, harmful emissions have to be controlled, and as indicated in Fig. 1, we have formed box plots for the emission gases in a grouped manner. We have considered the percentile count of the harmful emissions. NO_2 emissions are expected to be in the range of 21.5–33.4, while SO_2 emissions are in the range of 10.1–17.4, while O_3 is above 33.25–35. Thus, effective emission control strategies are required to be presented that controls the harmful emissions, while managing a stable load in GTS.

Thus, in this paper, we propose a novel scheme, *LEAPS* that presents the relationship between the offered load against the harmful emissions in GTS. In *LEAPS*, we propose an optimized route pattern, between given pair of start and stop points, and present the minimum number of stops that can serve the GTS passengers in a given area. Thus, the number of stops is reduced; the public vehicles can complete the trips at a low cost, and thus, harmful emissions can be minimized.

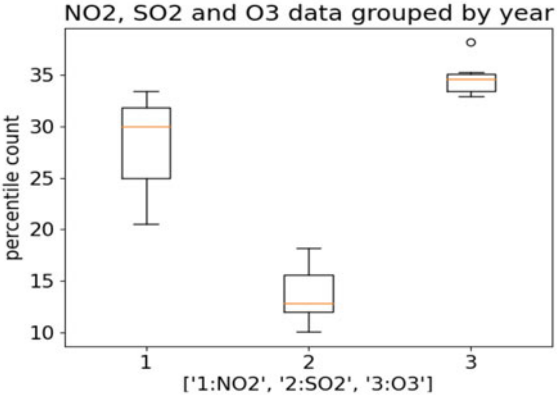


Fig. 1 Data of NO_2, SO_2 and O_3 grouped by state and year

1.1 Research Contributions

The following are the key contributions of this paper.

- We present the IoT-emission sensor-driven architecture of the *LEAPS* scheme that focuses on the route selection problem between a given pair of points.
- A route selection algorithm is proposed that minimizes the number of stop points of a public vehicle.
- This optimized route was then plotted on a real-world map. A highly optimized route compared to a traditional unregulated transport system was obtained.

1.2 Article Structure

The article is divided into five sections. Section 2 presents the existing state-of-the-art approaches for GTS. Section 3 presents the proposed scheme *LEAPS* that discusses the load and emission-driven performance characteristics for sensor-driven GTS. Section 4 discusses the performance evaluation of the proposed scheme, and finally, Sect. 5 concludes the paper.

2 State-of-the-Art

In this section, we present the comparative analysis of existing state-of-the-art schemes. The details are presented as follows.

2.1 Related Work

The section presents the details of state-of-the-art (SOTA) approaches in ITS. Table 1 presents a comparative analysis of our proposed approach *LEAPS*, with existing SOTA. Researchers mostly have explored the issues of loads on vehicles and route management but have not addressed the issues in coherence. For example, Elgarej et al. [2] fail to address the issues of the number of times a public vehicle stops at given designated points, and thus, the emission control strategy is non-optimal.

Authors in [12] proposed an optimized route for a reduction in greenhouse emissions, but the work does not discusses the impact of a number of bus stops. Umit et al. [6] present the discussion on the optimized route between given points but does not discuss the effect of emission characteristics. Hulagu et al. [7] do not quantify load on a given public vehicle. Authors in [8] present a meta-heuristic algorithm that accommodates the load aspect on buses, but the scheme does not discuss the impact

Table 1 State-of-the-art parameters for GTS

Authors	Year	Parameters				Key contributions
		a	b	c	d	
Elgarej et al. [2]	2017	Y	N	N	Y	A path-planning scheme that minimizes the transportation costs based on servicing constraints on distance, fuel, and route
Hattrage et al. [3]	2018	N	N	Y	N	The work proposed a global positioning system (GPS) tracking system based on long-range IoT transmission (LoRa), and a working prototype is presented
Huo et al. [4]	2019	Y	N	N	Y	A joint optimization on multi-point distribution flow problem is presented on SBRP problem
Ellegood et al. [5]	2019	Y	N	N	N	The SBRP problem is presented as sub-problems, and key characteristics like number of schools, mixed load, service environment, objective, and constraints are discussed
Umit et al. [6]	2019	Y	N	N	Y	The work aims at the reduction of bus purchase costs, and reduction of fuel costs by minimization of bus fleets
Hulagu et al. [7]	2020	Y	N	N	Y	The work formulates the exact solution of a vehicle routing problem with environmental concerns and formulates a flow-based mixed-integer linear program for the environment-friendly SBRP
Hou et al. [8]	2020	Y	N	N	Y	The work presents a multi-pick-up problem via a pick-up and delivery problem with time windows (PDPTWs), to address the mixed loads and focus on the minimization of the number of buses
Ladha et al. [9]	2020	Y	Y	Y	N	The work presents a novel scheme *IIGPTS* that measures emission sensor readings based on offered loads and sets up user traffic requests, and request drops to minimize the emission levels of a public transport system (PTS)
Chavhan et al. [10]	2021	N	N	Y	Y	A context-aware vehicle incidents route service management is proposed for the ITS, and the incident impact time and vehicle density are measured in the incident zone. The relevant routes are proposed based on the traffic density of a given route
Jiang et al. [11]	2021	N	N	Y	Y	An integration framework for ITS is proposed that minimizes carbon footprints through emission measurements
Proposed	2021	Y	Y	Y	Y	The work presents an IoT driven, intelligent routing-based approach to GTS that focuses on load reduction via minimization of several stops between a given pair of source and destination points.

[a] Load
[b] Emission
[c] IoT driven
[d] Intelligent route
Y—denotes the parameter is present
N—denotes the parameter is not present

on emissions. Authors in [3] focus on intelligent routing, with consideration on the emission characteristics, but the load is not considered. A significant improvement is presented by Ladha et al. [9] that presents a scheme that is IoT driven and measures the emissions for a given load. The scheme discusses the scheduling of stops based on traffic requests and measures how many total public requests are satisfied. However, the authors did not present an intelligent route map between the given pair of points.

2.2 Research Gap and Novelty

The proposed work *LEAPS* employs a sensor-driven scheme that monitors the harmful emissions and loads characteristics but has not addressed route selection and optimization. Thus, in the proposed scheme, we initially cluster the passengers to minimize the bus stops. The problem is solved through a combination approach proposed in SBRP [13, 14], where the authors proposed an effective schedule for a given fleet of school buses on a designated route. Students are picked up at several bus stops and transported to their allocated schools, with the mentioned constraints and restrictions. In the scheme, we model the optimal route approach on a map and also consider the impact of load and emission on the selected route. Thus, the proposed scheme essentially addresses the related gaps in the previous schemes by presenting a unified model that considers the route and emission problem as a coherent unit.

3 *LEAPS*: The Proposed Scheme

In this section, we present the proposed scheme *LEAPS*. Figure 2 presents the details of the IoT-driven reference scheme. In the given model, we consider a bus fleet $H = \{h_1, h_2, h_3, \ldots, , h_n\}$, and the average load for a given H, denoted as $L = \{l_1, l_2, l_3, \ldots, l_n\}$. The scheme is designed to reduce the number of stops so that the

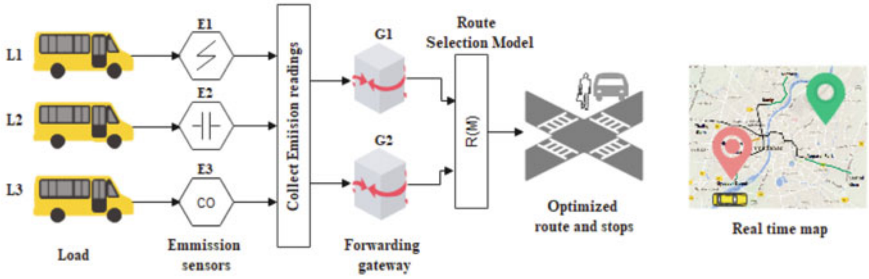


Fig. 2 *LEAPS*: IoT-driven reference scheme

harmful emissions such as CO_x and NO_x are minimized for a given run. We consider a bottom-up approach to measure the effect of GHG emissions of nth vehicle over a designated route R, with given start and end points. We install three sensor-units $\{S_1, S_2, S_3\}$ that are emission sensors, and they, respectively, measure the CO_x, NO_x, and O_3 unit. The impact of green house emissions, as presented in Ladha et al. [9], is considered. For any nth bus fleet h_n and given load l_n and fuel type f, the relation for emissions is presented as follows [9].

$$E_n = \sum_{f} \sum_{y=k-1}^{k} (s_{rq}^{v} \times (1 - \beta) \times Lk_e) \tag{1}$$

where E_n presents the emission value, s_{rq} denotes the bus fleet sales depending on fuel type f, in a particular qth year, and v represents the maximum sales units, β represents the running fraction of total fuel f over a day, and LK_e represents the load measurement over the emission type e. Based on the emission value, we consider the start point of a vehicle $S_p(x_p, y_p)$, and end point as $E_q(x_q, y_q)$. The goal is to install sensors units S over the vehicle, that collects the emission readings and sends them to the forwarding gateway units F_g. We consider two gateway units G_1 and G_2, for effective load balance. The emission data is sent to F_g through message queue telemetry transport (MQTT) protocol, that defines a message broker M_{br} and associated client units $\{h_1, h_2, \ldots, h_n\}$. M_{br} collects the readings from F_g, and performs routing to forward the data to destination nodes. The client units publishes the topics t_q an sends the control message c_q, t_q to M_{br}.

The published topics t_q are subscribed by clients, which is presented as follows

$$T = \{E_R, h_{\mathrm{id}}, R_{\mathrm{in}}, T\} \tag{2}$$

where E_R represents the emission readings, h_{id} represents the vehicle identifier, R_{in} represents the route information based on source and destination points, and T represents the timestamp of published topic. Any user U, who wishes to move through GTS, has to select appropriate h_n and is issued a ticket $T(h_n)$. The details of the ticket are depicted as follows

$$T(H_n) = \{U_{\mathrm{id}}, A_{\mathrm{info}}, (S_p, E_q), B_T\} \tag{3}$$

where U_{id} denotes the user identifier, A_{info} denotes the application meta information from where the ticket is booked, and B_T denotes the booking timestamp. Based on TH_n, U is associated with a bus-stop B_{st} for on-boarding the bus. In this manner, the load L on the bus is fixed, and once the bus reaches its optimal load carrying capacity L_{max}, the starting coordinates of the first stop are fed to the system. We present the coordinates as two-dimensional maps $S\{x_p, y_p\}$ and $D\{x_q, y_q\}$. We compute the Euclidean distance between the two points. The obtained coordinate sets $\{(x_1, y_1), (x_2, y_2), \ldots, (x_i, y_i), (x_{i+1}, y_{i+1}), \ldots, (x_n, y_n)\}$ are sorted in ascending order, and equidistant plots are generated. We denote the equidistant point set

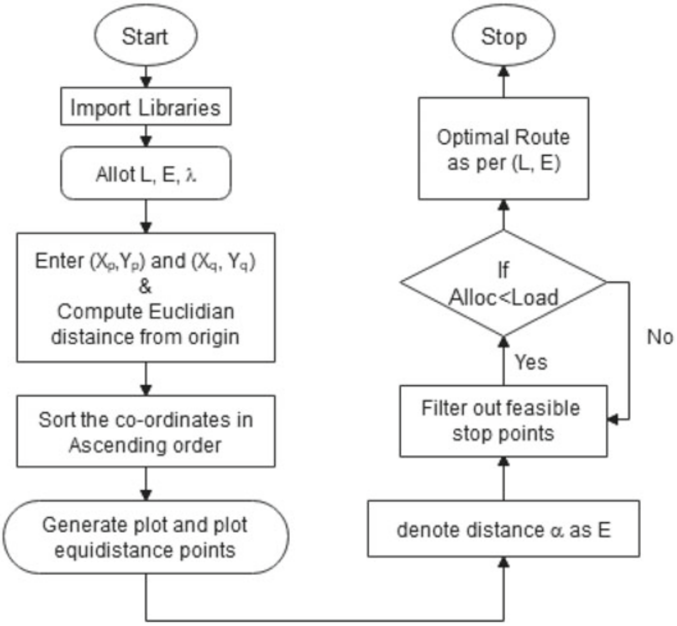

Start
Import Libraries
Allot L, E, λ
Enter (Xp,Yp) and (Xq, Yq)
&
Compute Euclidian distaince from origin
Sort the co-ordinates in Ascending order
Generate plot and plot equidistance points
denote distance α as E
Filter out feasible stop points
Yes
If Alloc<Load
No
Optimal Route as per (L, E)
Stop

Fig. 3 *LEAPS*: the operational flow of the proposed scheme

as E_P and present an optimized route algorithm R that minimizes B_{st} to minimize emissions E. Figure 3 presents the operational flow of the proposed scheme. In the proposed scheme, we have assumed that the first B_{st} is at origin (0, 0) for simplicity, and the route predicts the order at which users are picked up. E_q is plotted such that we present the potential bus stops so that each user have to travel same distance d to reach B_{st} on an average, when the stops are minimized. The feasible user-stop pairs (U, B_{st}) are filtered out, and remaining B_{st} are eliminated in case no U is allocated to them. The process is repeated in iterative fashion until all the feasible (U, B_t) pairs are returned. The generated B_{st} are then plotted on real-world map.

Algorithm 1 presents the details of the route optimization algorithm. In the proposed algorithm, lines 1–4 initialize the source at source S_p and destination E_q, and under the worst-case assumption where all users U are mapped to single B_{st}, and the vehicle bus is thus mapped to maximum capacity through the $MaxCap$ variable. Next, we enter the start and end points in lines 5–8. Lines 9–12 check the requirements of user stops B_{st}, and values are users stops are entered. Lines 14–22 use the Euclidean distance to find the minimum distance between the start points S_p and bus-stop arbitrary point (x_h, y_h). Once points are obtained, we sort the points to minimize the number of stops; in case of less users are at a particular stop, we present them with the nearest stop locations. Lines 24–35 depict the scenario. Lines 36–43 present the list of final mapped users and restricted stop points, and lines 44–46 plot the points to form the optimized route R. Assuming there are n points in

Algorithm 1 *LEAPS*: The route optimization algorithm

Input: Required Libraries L_{req}, Load L, Equidistant Points E_p, Start point $S_p(x_p, y_p)$, and End-point $E_q(x_q, y_q)$
Output: Optimized route R based on minimization of emissions E
Initialization: L=100, E_p=25

```
procedure ROUTE_SETUP($S_p$, $E_q$, $L_{req}$)
    $S_n \leftarrow 0$
    Enter $(x_p, y_p)$ and $(x_q, y_q)$
    $MaxCap \leftarrow ([0] * 2) \times L$
    $MaxCap[n] \leftarrow (x_p, y_p)$
    $n \leftarrow n + 1$
    $MaxCap[n] \leftarrow (x_q, y_q)$
    $n \leftarrow n + 1$
    while ($ch \neq 1$ and $Stops_{num} \leq 1$) do
        Enter $(x_h, y_h)$
        $n \leftarrow n + 1$
        REPEAT UNTIL no more $(x_n, y_n)$ are entered
    end while
    $i \leftarrow 0$
    $j \leftarrow 0$
    $d \leftarrow [0]$
    while ($i < n$) do
        Print ($C[i][0]$, $C[i][1]$)
        $d[i] \leftarrow math.sqrt\ (pow(C[i][0], 2) + pow(C[i][1], 2))$
        Print $d[i]$
        $i \leftarrow i + 1$
    end while
    Print $d$ and $C[i][j]$
    $Sort(d) \leftarrow d.sort()$
    $S_c \leftarrow [[0] * 2] * n$
    while ($i < n$) do
        $j \leftarrow 0$
        while ($j < n$) do
            $p[j] \leftarrow math.sqrt\ (pow(C[j][0], 2) + pow(C[j][0], 2))$
            if ($p[j] == d[j]$) then
                $p[j] = d[j]$
                $j \leftarrow j + 1$
            end if
            $i \leftarrow i + 1$
        end while
        $x = ([0] * 1) * n$
        $y = ([0] * 1) * n$
    end while
    while ($i < n$ do
        $x[i] \leftarrow S[i][0]$
        $y[i] \leftarrow S[i][1]$
        $i \leftarrow i + 1$
    end while
    Plot points from $(x_p, y_p)$ to $(x_q, y_q)$
    $\alpha = np.linspace\ (x, y)$
    Print final optimized route $R$
end procedure
```

the ecosystem, the worst-case scenario would have users in all the stops with load L. Thus, the time complexity of the algorithm is $O(n^L)$. The n points are stored as vectors in array, and thus, the space complexity is $O(n)$.

4 *LEAPS*: Performance Evaluation

In this section, we present the performance evaluation of the proposed scheme *LEAPS*. We start with the experimental setup and simulation parameters. Table 2 presents the details of the simulation parameters.

4.1 Experimental Setup

In experimental setup, we have considered $MH - Z16$ for measurement of CO_2 emissions, and NO_2 sensor $MikRoe$. We used Raspberry Pi to build the server. For connectivity, a GSM module is set up. The simulation plots are prepared in Python with the following libraries.

- *matplotlib.pyplot*: To plot the initial student locations, unoptimized route, and potential B_{st} on the unoptimized route and final optimized route in a 2D coordinate system.
- *scipy.interpolate* : For interpolation between set of users and locations, and the library plots equidistant points which served as potential B_{st}. Later, B_{st} with no users are dropped and are followed by the candidate bus stops which provide a sub-optimal solution. The final set of stops is an optimized solution (sub-set) of these interpolated points.
- *folium*: This library is used to visualize geospatial data. The final route along with bus stops has been plotted on a real-world map.
- *Openrouteservice* : The *openrouteservice* library provides us with access to the open-route service (ORS) application programming interfaces (APIs).
- *Pandas* : Read a .csv file that contains geographical coordinates of stops in a sequential manner. The read data are stored in a data frame, which then is used to plot the actual route of a bus.

4.2 Simulation Results

Now, we present the details of the simulation results.

Table 3 presents the outputs of Euclidean stop points based on user-location coordinates. In this L represents the load conditions, λ represents the maximum distance that any user U takes to reach B_{st}, and E_p denotes the set of equidistant points.

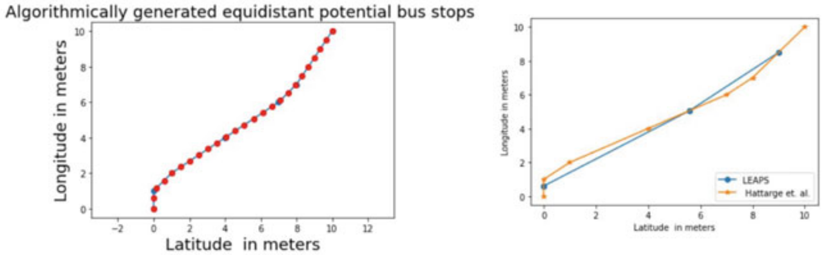
Figure 4a presents the details of the generated stops by the route optimization Algorithm 1. The algorithm determines the relative position of stops on a real-world map, and geographical coordinates are stored in .csv file. The file is imported into code that generates the map-points.

Table 2 Simulation parameters

Parameter	Values
L	1–10
E_p	25
Initial S_p	(0, 0)
h_n	4
Gateways G	2
Sensors S	3
β	0.8
λ	2–3

Table 3 Stop point locations based on user-location coordinates

User coordinates	L	λ	E_p	Stop locations (Euclidean)
(1, 2), (8, 9), (2, 3), (3, 4), (2, 8), (4, 0)	6	2	25	(1.5700676785484586, 2.5700676785484586), (3.653218997575661, 1.3871240096973556), (5.614315224004558, 8.602385870667426)
(0, 0), (1, 2), (1, 1),\ (4, 4), (10, 10),(8,7), (7, 6)	7	2	25	(0.43148483548232186, 0.43148483548232186), (5.591701682664247, 5.0611344551094986), (8.984546340333662, 8.476819510500494)
(0,0), (1,2), (1,1),(4,4), (10, 10),(8,7), (7, 6)	7	2	20	(0.5450334763987223, 0.5450334763987223), (2.839287815673911, 3.226191877115941), (6.68732273651477, 5.791548491009847), (9.14488112870203, 8.717321693053046)
(0, 0), (1, 2), (1, 1), (4, 4), (10, 10),(8,7), (7, 6)	7	3	25	(0.43148483548232186, 0.43148483548232186), (5.591701682664247, 5.0611344551094986), (8.984546340333662, 8.476819510500494)



(a) Plotted B_{st} points between start and end points

(b) Comparative analysis of lower stops to minimize emissions

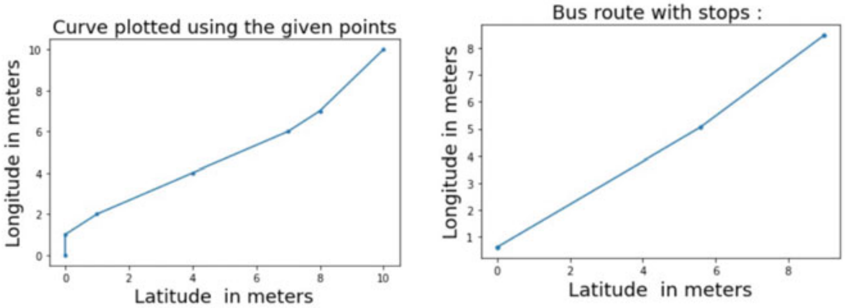
Fig. 4 Comparative analysis of the proposed route optimization algorithm in lowering emissions

Figure 4b draws a comparison between the proposed scheme against the scheme presented by Hattrage et al. [3]. Authors in [3] have not employed intelligent routing, and thus, the effective number of stops is higher in the scheme. In comparison, our scheme employs effective scheduling and stops management that minimizes the stops, and thus makes the route selection shorter, that reduces the emission footprints.

Next, we present the viability of the optimized routing algorithm. For simulation purposes, we have considered that 7 users are to be picked up from their respective locations. Figure 5 presents the plot of the users from their initial location. Once, we execute the algorithm, we optimize the conditions that instead of stopping the bus for 7 times, we have to make only 3 B_{st}. Figure 5 presents the optimized results. The reason is that some stops would be eliminated as the users would shift to alternate stops. Thus, users are clustered into a single stop, which minimizes the number of stops. We have also considered that load L of the vehicle is at maximum; an optimal set of stops are presented as output. Also, in case a user moves to a particular stop, we consider E_q to make the distance equidistant, where the distance between the user initial location S_q and alloted stop is always bounded by a parameter λ, which depicts walkable distance.

Finally, we present the relationship between the load of the individual bus at any arbitrary stop and the GHG emission. We consider three scenarios—fewer stops with many students at each stop, optimized number of stops, stops with algorithms, and worst case where each student is at a particular stop. Figure 6 presents the scenarios.

1. *First scenario*—this scenario represents the maximum number of stops, where every user is at a single stop, This is the worst-case scenario with high GHG emissions as bus stops are maximized.
2. *Second scenario*—it represents the optimal number of stop requirements as per our optimal route algorithm presented in *LEAPS*, with the same offered load. As indicated, the GHG emissions have decreased significantly.
3. *Last scenario*—this represents the ideal scenario, where we have the least number of stops. In such cases, the load at a particular stop would increase drastically, and equidistant conditions would not be followed for all users, and thus, some users



(a) Stops for all passengers before algorithm execution (b) Stops for all passengers after algorithm execution

Fig. 5 Benefits of the proposed route optimization algorithm in lowering emissions

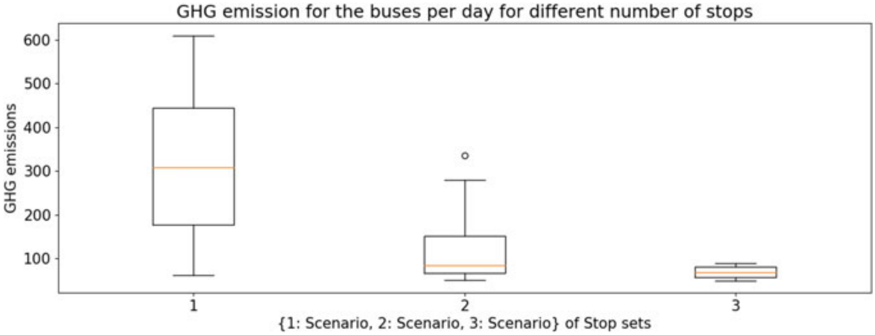


Fig. 6 Emission comparison with respect to types of stops in the aspect of different buses

have to travel a higher distance to reach a particular stop, compared to others who might have to travel less. However, this scenario would have the lowest GHG value, but in practical deployments, this reduces the quality of offered service to users.

5 Conclusions

Modern IoT ecosystems are environment-friendly, and thus, the focus has shifted toward the design of effective GTS that can address the dual challenges of mitigation of harmful emissions, management of passenger loads, and effective route setups. In a similar direction, in this paper, we propose *LEAPS*, which presents a scheme that optimizes the GTS route setup between start and end stations in respect of passenger load and emission characteristics. We consider a dynamic route mechanism that manages the route according to passenger load at designated stops. Those stops which have few passengers are not considered in the dynamic route setup, and the passengers are notified before route changes. This averages the loads at all the stops, and thus, the route map between the designated set of points is optimized. We also measure the relationship between the offered GTS load at a particular stop and the impact on overall emission rates.

As part of the future scope, the authors would propose deep learning strategies for route management, where routes can be managed effectively based on lane selection. We would analyze the input traffic distribution, and weather conditions as inputs to the learning model, to predict the route selection map, and consequently, the impact on load and emission characteristics.

References

1. Transport and carbon dioxide emissions: forecasts, options analysis, and evaluation. http://www.indiaenvironmentportal.org.in/files/Transport-CO2-Emissions.pdf. Accessed 26 Aug 2019
2. Mouhcine E, Khalifa M, Mohamed Y (2017) Route optimization for school bus scheduling problem based on a distributed ant colony system algorithm. In: 2017 intelligent systems and computer vision (ISCV), pp 1–8
3. Hattarge S, Kekre A, Kothari A (2018) Lorawan based GPS tracking of city-buses for smart public transport system. In: 2018 first international conference on secure cyber computing and communication (ICSCCC), pp 265–269. https://doi.org/10.1109/ICSCCC.2018.8703356
4. Huo L, Yan G, Fan B, Wang H, Gao W (2014) School bus routing problem based on ant colony optimization algorithm. In: 2014 IEEE conference and Expo transportation electrification Asia-Pacific (ITEC Asia-Pacific), Beijing, China, pp 1–5. https://doi.org/10.1109/ITEC-AP.2014.6940973
5. Ellegood WA, Solomon S, North J, Campbell JF (2020) School bus routing problem: contemporary trends and research directions. Omega 95:102056. https://doi.org/10.1016/j.omega.2019.03.014
6. Ümit U, Kilic F (2019) A school bus routing problem using genetic algorithm by reducing the number of buses. pp 1–6. https://doi.org/10.1109/ASYU48272.2019.8946425
7. Hulagu S, Celikoglu HB (2020) Environment-friendly school bus routing problem with heterogeneous fleet: a large-scale real case. IEEE Trans Intell Transp Syst 1–11 (2020). https://doi.org/10.1109/tits.2020.3036696
8. Hou YE, Dang L, Dong W, Kong Y (2020) A metaheuristic algorithm for routing school buses with mixed load. IEEE Access 8:158293–158305. https://doi.org/10.1109/access.2020.3019806
9. Ladha A, Bhattacharya P, Chaubey N, Bodkhe U (2020) IIGPTS: Iot-based framework for intelligent green public transportation system. In: Singh PK, Pawłowski W, Tanwar S, Kumar N, Rodrigues JJPC, Obaidat MS (eds) Proceedings of first international conference on computing, communications, and cyber-security (IC4S 2019). Springer, Singapore, pp 183–195
10. Chavhan S, Gupta D, Nagaraju C, Rammohan A, Khanna A, Rodrigues JJPC (2021) An efficient context-aware vehicle incidents route service management for intelligent transport system. IEEE Syst J 1–12. https://doi.org/10.1109/JSYST.2021.3066776
11. Jiang J (2021) Intelligent city traffic scheduling optimization based on internet of things communication. Wireless Commun Mob Comput 2021
12. Newton RM, Thomas WH (1969) Design of school bus routes by computer. Socio-Econ Plan Sci 3(1):75–85. https://doi.org/10.1016/0038-0121(69)90051-2

13. Bögl M, Doerner KF, Parragh SN (2015) The school bus routing and scheduling problem with transfers. Networks 65(2):180–203. https://doi.org/10.1002/net.21589
14. Park J, Kim BI (2010) The school bus routing problem: a review. Euro J Oper Res 202(2):311–319. https://doi.org/10.1016/j.ejor.2009.05.017

A Smart Home Automation System Based on Internet of Things (IoT) Using Arduino

Mohini Darji, Naina Parmar, Yashesh Darji, and Shivangi Mehta

Abstract In recent years, home automation has grown in popularity. Its goal is to assist people in managing their home appliances independently and creating a self-sufficient environment at home. The goal of this paper is to create a home automation system to manage household appliances via wireless communication such as Wi-Fi. This smart home system was designed with the implementation of relevant software and hardware. Passive infrared sensor (PIR) is used to sense the motion and according to that light will on or off. Each room's temperature is monitored and maintained at room temperature using a temperature sensor that operates a fan powered by a DC motor to keep the temperature stable. Ultrasonic distance sensor is used to detect the distance according to that micro servo motor open or closed the door. The Arduino uno is utilized for these control applications because the Arduino has the advantages of being simple to comprehend and modify. The Arduino board is a circuit board specifically built for programming and prototyping with ATMEL microcontrollers. The microcontroller in this Arduino is an ATmega 328, which comes pre-installed on the board. The suggested framework is both cost-effective and extensible, as it allows several devices to be connected and controlled.

Keywords Arduino Uno R3 · Temperature sensor · Ultrasonic distance sensor · Dc motor · LCD 16*2 · Resistors · PIR Sensor · Potentiometer · Positional micro servo · H-Bridge motor driver

M. Darji (✉) · N. Parmar · S. Mehta
Department of Computer Science and Engineering, Devang Patel Institute of Advance Technology and Research (DEPSTAR), Charotar University of Science and Technology (CHARUSAT), Changa, Gujarat, India
e-mail: mahi.darji1992@gmail.com

N. Parmar
e-mail: nainaparmar.dcs@charusat.ac.in

S. Mehta
e-mail: shivangimehta.dcs@charusat.ac.in

Y. Darji
Department of Mechanical Engineering, Rai School of Engineering, Rai University, Ahmedabad, Gujarat, India

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_19

1 Introduction

IoT can be characterized as an innovation wherein certified actual components or contraptions with data distinguishing, planning, and self-gathering capacities can be utilized to work together with other comparable gadgets and interaction that information to settle on an educated choice that will be helpful in our everyday lives. Human exertion is extraordinarily limited in the Home Automation System. In view of progressions in home computerization, human life is overseen without trouble. The internet of things has risen to control and monitor various equipment such as fans, bulbs, air conditioners, televisions, and washing machines, and it is driving human living with serenity. Because many sensors and electrical devices are made by different companies, there may be an increase in interoperability concerns. Because all sensors and devices are connected via the internet, the IoT can overcome interoperability concerns [1].

Home automation provides residents with a sense of security. As a result, lights in storerooms, stairwells, and other dark areas are turned on. As a result, the chances of stumbling or running into something are reduced. Internet access is used in home automation to control devices from afar. For a long time, the Internet was mostly used for surfing the web, searching for data, and downloading software and other items. The pace of innovation is propelling the Internet's collaboration with technology and gadgets. The comfort and security of homes have been increased thanks to the home automation framework. Individuals are also concerned about expenditures. In the workplace, a distinct group of people is assigned to supervise certain manual means written job. Those ambitions are being replaced by home automation. The cost is significantly reduced as a result of this.

In present era, home automation technology is a fast increasing technology that allows users to modify their houses to the point where they can execute a variety of chores automatically and autonomously. As new features are implemented to meet the requirements, this phase will gradually evolve. The goal of this operation is to cut down on energy usage. These gadgets assist individuals in leading healthy lifestyles and conserving energy for future demands. These systems are designed to monitor and control a wide range of networked appliances, including lighting, temperature, and doors. Home automation will be one of the hottest fields in the near future. Furthermore, the sector for home automation systems in the future will be extremely diverse.

2 Motivation

In certain circumstances, there may be handicapped people in the house who are unable to move around regularly to manage the appliances. Yet, these people may simply control all of the appliances utilizing a home automation system. It is critical to build home automation systems that require minimal and simple user involvement for handicapped people. Home automation systems also raise the level of living by providing a user interface that is simple, versatile, and engaging. We need to use modern technologies and equipment to give all functionalities in a low-cost and flexible setting.

Section 2 gives detail survey of home automation system. Detail about different devices is discussed in Sect. 3. Section 4 presents proposed design and experimental setup. Section 5 presents conclusion.

3 Related Works

There have been many researchers who have attempted home automation system. In this section, we discuss about related work regarding home automation system.

Tejesh and Neeraja [1] created and implemented a framework in which data from rooms 1 and 2 is collected in the Raspberry Pi 3 and then displayed on the site page, and if a crisis situation arises, the user will receive an SMS to make the best decision. The designed technology can be used not only in the lab but also in real-time scenarios. In [2] Satapathy et al., present a minimal expense, adaptable, and trustworthy home mechanization framework with added security dependent on the Arduino microcontroller and IP availability through nearby Wi-Fi, which permits approved clients to view and control gadgets distantly utilizing smartphone applications.

Mihalache [3] introduced a home mechanization system dependent on Arduino Uno and significant modules, which takes into consideration the control of lights or fans, with changes made relying upon information from different sensors. The framework was designed to be low-cost and extensible, while also providing availability, comfort, and energy efficiency. S. Singh and K. Ray introduced an outline of the Internet of Things, models, and fundamental advancements, just as their applications in our regular daily existences, in [4]. Home automation is becoming a reality thanks to the Internet of Things, and firms like Apple, Amazon, Google, and Samsung are collaborating to create the platform and solutions for smart homes.

The utilization of Raspberry Pi and IOT innovation to take advantage of a home administration and security framework was presented by Pavithra and Balakrishnan [5]. The framework can be used for ongoing home security observing, distant administration of homegrown apparatuses, and fire anticipation with fast reactions. The framework may be used in an assortment of settings, like banks, emergency clinics, and labs, to definitely diminish the shot at unapproved section. Proof of theft may be given to the security department if there is a problem. According to Abdulraheem et al. [6], the usage of IoT has smoothed out the presentation and automation of house and building gadgets and machines, ensuring that they deliver powerful and effective defense, relaxation, and comfort for a great cause.

Singh et al., [7] fostered a home robotization framework dependent on IOT utilizing a Wi-Fi based microcontroller. This examination has brought about the improvement of a constant home robotization framework that is both utilitarian and mechanically progressed. R. Kodali and S. Yerroju built a low-cost, low-power ESP8266 embedded Wi-Fi module to control a transfer channel that fills in as a change to control home devices in [8]. They used Thinger .io's cloud infrastructure, which allowed the smartphone and the ESP8266 to connect over REST API.

Nitu et al. [9] developed and implemented a multi-functional sophisticated home automation system based on Android and the web. The suggested framework ensures that household appliances can be controlled easily and effectively from anywhere on the planet. By preventing intruders, this structure improves the security of a residence. It also protects a home from disaster by preventing gas spills and fire suffocation. This system is especially beneficial for people of senior age, those who are disabled, and people who are working. In [10], Mandula et al. offered two models: house mechanization using Bluetooth in an enclosed area and home computerization using ethernet in an outdoor environment.

Solunke et al. [11] S. Somani, P. Solunke, S. Oke, P. Medhi Using an Android smartphone, I was able to control household appliances with ease while also assuring robust home privacy and stability. Incorporating the voice call capability into the same smartphone application that the consumer uses to control his consumer electronics could improve the system in the future. The pros and disadvantages of various methodologies employed in home automation systems were reviewed in [12] by Sivapriyan et al. They looked at how bluetooth-based home automation can be built for a low cost and with a lot of flexibility, but it can only work inside the bluetooth wireless network's restricted range. The benefits of an IoT-based home automation system are greater, but there are certain disadvantages, such as the fact that it can only function in the presence of the Internet.

Mahalakshmi and Vigneshwaran [13] created a one-of-a-kind Android-based home management and monitoring system. The proposed architecture uses a tiny web server and bluetooth communication as an interoperable application layer to connect the remote user and the home gadgets. To access and manage home devices, any Android-based smartphone with built-in Wi-Fi connectivity can be utilized. When Wi-Fi isn't accessible, mobile cellular networks like 3G and 4G can be used instead, eliminating the need for an external voice recognition module. K. Venkatesh, P. Rajkumar, S. Hemaswathi, and B. Rajalingam showed a smart house automation prototype using IoT in [14]. This work will be continued in the real world by attaching relays to a Raspberry Pi board for controlling home appliances from afar. In addition, the authors provide a basic IoT framework that uses cloud computing infrastructure to connect and manage IoT devices. The use of smart home technology to promote family safety, particularly in relation to fire prevention and carbon monoxide monitoring, is projected to gain traction in the near future. A limited number of gadgets in home appliances can now be linked and controlled.

T. Sehgal, Shubham, and others utilized IoT and a portable application to execute home robotization in [15]. After client confirmation, this study creates and executes a web-based savvy home framework that can be controlled from a distance. The Android-based savvy home programming speaks with the minuscule web-server through the web utilizing a REST-based web administration. Any Android gadget can be utilized to control and screen the savvy home climate on account of the brilliant home programming. Since the microcontroller plays out all handling, a minimal expense shrewd home framework has been constructed that doesn't need a PC. The framework additionally utilizes Google's discourse acknowledgment motor, so it needn'tsssss bother with a different voice acknowledgment module. Future improvements could incorporate fusing SMS and call notices, just as decreasing wiring changes for introducing the suggested framework in previous houses by building a remote organization inside the home climate for controlling and observing the savvy home climate.

In [16] Shinde et al. proposed that a smart home system connects many types of house and electrical products, such as windows and fans. It allows customers to operate and use appliances according to their preferences. Following a review of various current systems, they offered a novel technique for improving human interaction and maximizing the use of Android and Arduino. Their approach can manage cost-effective, flexible, and energy-efficient smart homes with the help of a home automation system.

4 Hardware Description

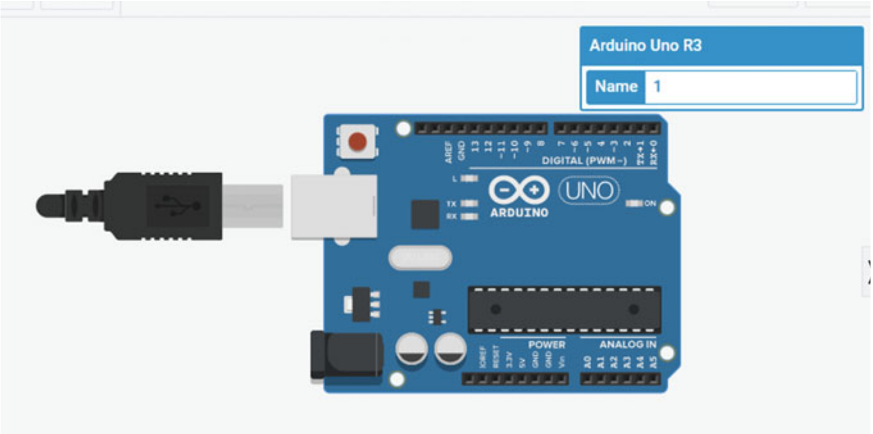
IoT equipment covers an assortment of gadgets like directing gadgets, extensions, sensors, etc. Framework initiation, security, activity particulars, correspondence, and identification of help explicit objectives and activities are only a couple of the significant obligations and capacities oversaw by these IoT gadgets. In this section, we discuss about different IoT hardware and sensors such as Arduino Uno R3, Temperature Sensor, Ultrasonic Distance Sensor, PIR Sensor, LCD 16*2, Potentiometer, Positional Micro servo, and H- Bridge Motor driver.

- **Arduino Uno R3**

The Arduino Uno is a microcontroller board based on the AT mega 328, as seen in Fig. 1. It has 14 digital input/output pins, 6 analogue inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button [13]. Table 1 lists a handful of the Arduino Uno's most important features.

- **Temperature Sensor**

A temperature sensor is used to estimate temperature via electrical indications, as shown in Fig. 2. Temperature sensors are made up of two metals that produce electrical voltage or blockage when exposed to changes in temperature. They are necessary for keeping a certain temperature in any system that produces anything from medicine to beer.



Arduino Uno R3

Name 1

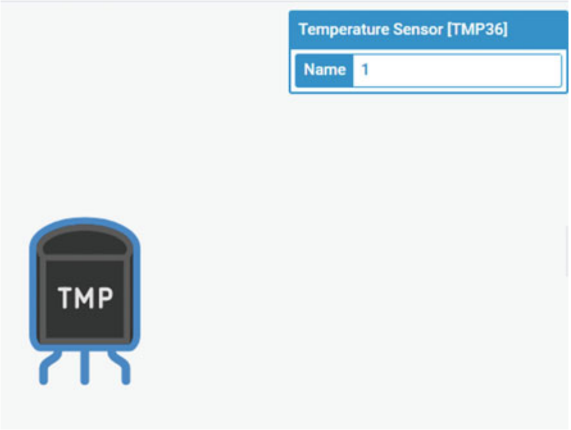
Fig. 1 Arduino uno board

Table 1 Arduino uno features

Items	Specification
Microcontroller	Atmega328
Crystal oscillator	16 MHz
Operating voltage	5 V
Input voltage	5-12 V
Digital I/O pins	14 (D0 to D13)
Analog I/O pins	6 (A0 to A5)
PWM pins	6 (pin # 3, 5. 6, 9, 10 & 11)
Power pins	5 V, 3.3 V, Vin, GND
Communication	UART (1), SPI (1), I2C (1)
Flash memory	32 KB (0.5 KB is used by bootloader)
SRAM	2 KB
EEPROM	1 KB

- **Ultrasonic Distance Sensor**

An ultrasonic sensor is an electronic gadget that identifies the distance between two articles utilizing ultrasonic sound waves and converts the reflected sound into an electrical sign. Ultrasonic waves travel at a quicker rate than perceptible sound waves (for instance the sound that individuals can hear). The transmitter (which creates sound utilizing piezoelectric stones) and the recipient (which recognizes sound after



Temperature Sensor [TMP36]

Name 1

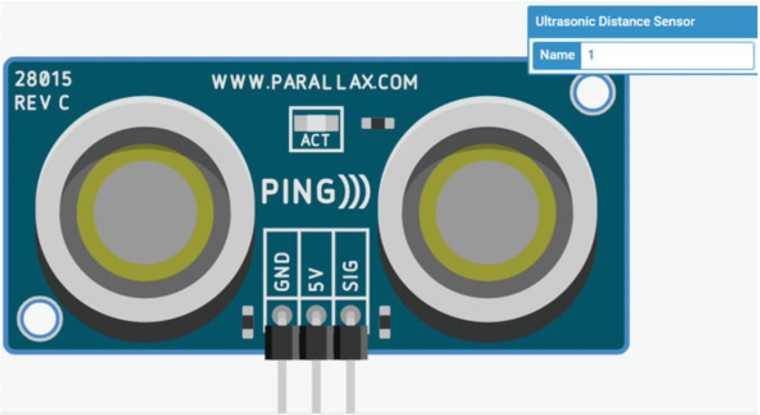
Fig. 2 Temperature sensor

it has headed out to and from the objective) are the two essential parts of ultrasonic sensors (displayed in Fig. 3).

- **PIR Sensor**

PIR sensors, as seen in Fig. 4, identify movement and are essentially consistently utilized to decide whether a human has entered or left the sensor's reach. They're little, modest, low-power, easy to utilize, and they don't wear out. PIR movement sensors are otherwise called "Aloof Infrared," "Pyroelectric," or "IR movement" sensors.

The exact information could be determined by light, heat, movement, moisture, pressure, or any of a variety of other natural features. The result is typically a signal

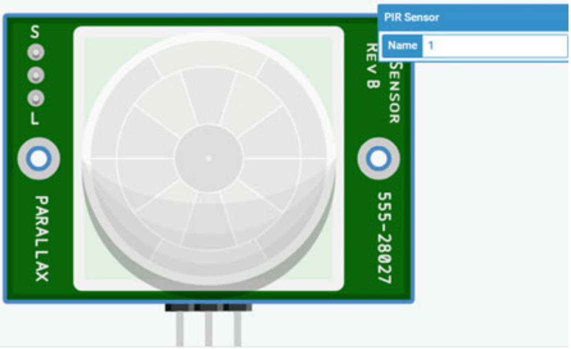


Ultrasonic Distance Sensor

Name 1

Fig. 3 Ultrasonic distance sensor

Fig. 4 PIR sensor



that is intended for an understandable display at the sensor region or routed through an organization for review or additional processing [13].

- **LCD 16*2**

A good example is the liquid crystal display (LCD) seen in Fig. 5. It's a type of electronic presentation module that's found in a variety of circuits and devices, including cell phones, minicomputers, PCs, and televisions. The most well-known applications for these showcases are multi-portion light-radiating diodes and seven fragments. The main advantages of using this module are its low cost, ease of programming, liveliness, and the fact that there are no restrictions on displaying new characters, remarkable and even movements, and so on.

- **Potentiometer**

A potentiometer is a three-terminal, physically customizable variable resistor. The finishes of a resistive component are associated with two terminals, while the third terminal is associated with a movable wiper; resistive divider ratio is determined by

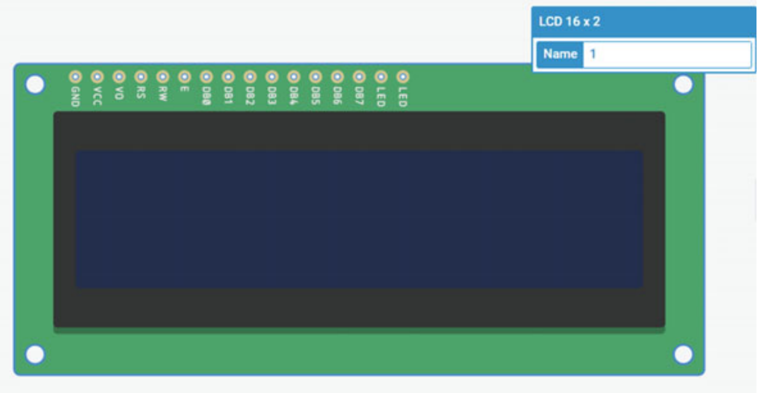
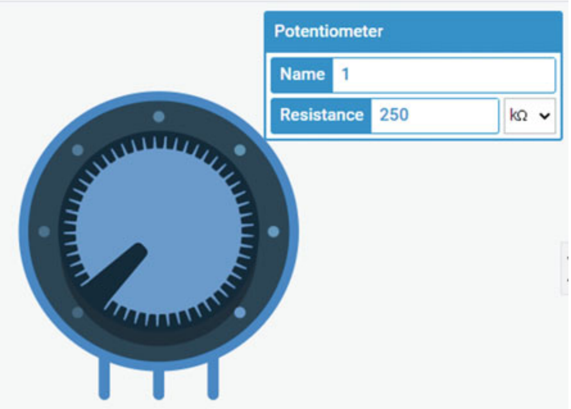


Fig. 5 LCD 16*2


Potentiometer

Name 1

Resistance 250 kΩ

Fig. 6 Potentiometer

the position of the wiper. By appropriately measuring voltage, they assist in obtaining a variable voltage from a fixed-voltage source. They work without the use of energy or additional circuitry because they are passive devices. Figure 6 shows a potentiometer in action.

- **Positional Micro Servo**

A servomotor is a shut circle servomechanism that controls its movement and extreme position utilizing position input. The position instructed for the yield shaft is addressed by a sign (analogue or digital) that is taken care of into the control (Fig. 7).

To give position and speed criticism, the engine is associated with a position encoder. Just the position is estimated in the most fundamental situation. The deliberate yield position is contrasted with the order position, which is the regulator's outside input. In the event that the yield position doesn't coordinate with the ideal position, a mistake signal is produced, making the engine pivot one or the other way to carry the yield shaft to the right position. The blunder signal abatements as the positions approach zero, and the engine stops.

- **H-Bridge Motor Driver**

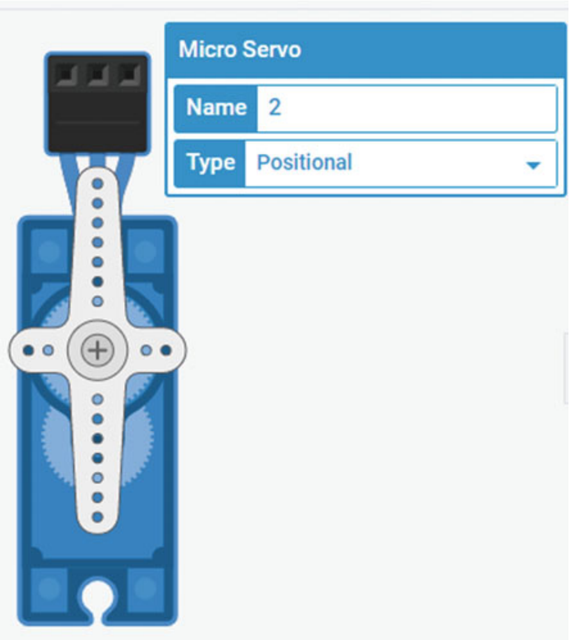
The H Bridge is a basic electronic circuit that allows us to apply voltage to a load in either direction (see Fig. 8). It is often used to control DC motors in robotics applications. The H Bridge allows us to run a DC motor in either a clockwise or anticlockwise direction. This circuit is also used in inverters to generate alternating waveforms.

- **BreadBoard**

The breadboard (Fig. 9) is a device that uses male-to-male wires to connect electronics components conveniently and without soldering (with pins at both ends) [1].

A breadboard is used to quickly construct and test circuits before completing any circuit design. Circuit components such as ICs and resistors can be added through the

Fig. 7 Positional micro servo

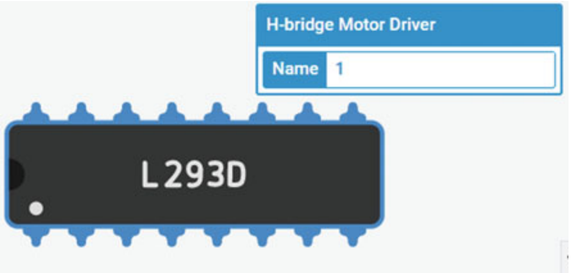


Micro Servo

Name 2

Type Positional

Fig. 8 H-Bridge motor driver



H-bridge Motor Driver

Name 1

many holes on the breadboard. Metal strips run underneath the breadboard and join the slots on the top. Horizontally, the holes in the top and bottom rows are connected, while vertically, the holes in the middle and bottom rows are connected.

To utilize the breadboard, the legs of the components are put into the holes. A node is made up of a series of holes that are connected by a metal strip beneath them. A node is a connection point in a circuit between two components. The legs of different components are connected by putting them in the same node. Connections to the power source are usually made through the long top and bottom rows of holes. Assemble the components and link them with jumper wires to complete the circuit. On one side of the main line, half of the legs are on one side, and the other half are on the other. The ICs are placed in the middle of the board.

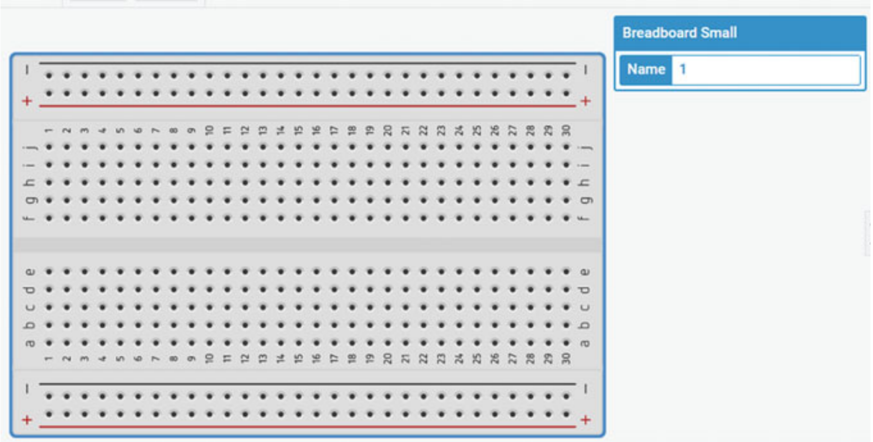


Fig. 9 Bread board

5 Proposed Design and Experimental Setup

5.1 Proposed Design

Figure 10 depicts the proposed Home Automation System design. It consists different devices such as sensors like temperature, ultrasonic distance, PIR and rest of devices are Arduino Uno R3, Breadboard, potentiometer, DC motor, positional micro servo, lcd 16*2, and H-Bridge motor driver.

It's a home automation device where-(i) The door will open if everybody comes close to on the door inside 40 cm and door can be open for two seconds. Then it'll test once more if everybody remains inside 40 cm, if yes, then the door will nevertheless

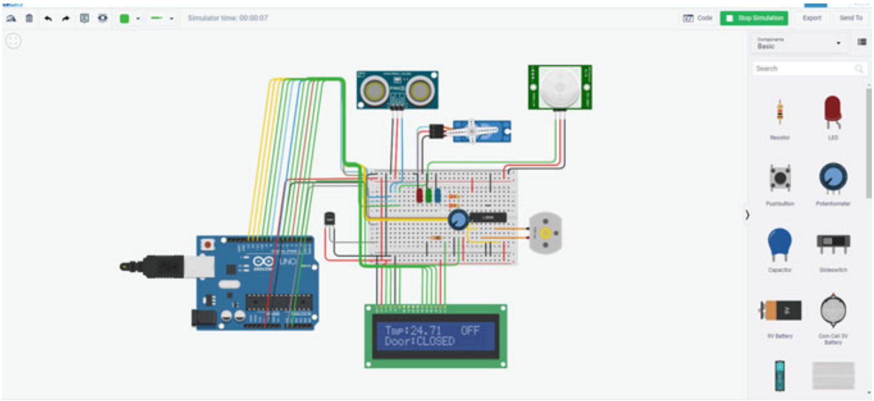


Fig. 10 Home automation system based on IoT using arduino uno

open for two extra seconds, and if no, then the door will mechanically be closed. For that we used right here USD for measuring Servo motor for commencing the door and distance. (ii) detects any motion in the room, mild (LED) will mechanically be lighting. If there's no motion withinside the room, then the mild will continue to be off. For that, we used right here PIR sensor for detecting motion and LED light. (iii) It detect room temp. and, either if it is more than 24 °C, a fan activated; nor remain turned off. We utilized a temp. sensor LM35 for temp. detection and a motor to jog a fan.

5.2 Experimental Setup

In this section, we explored how to use the TINKERCAD simulator to construct a home automation system. TINKERCAD [25] is a fantastic program for simulating Arduino-based devices and much more. Before doing any workouts or even your own inventions on real hardware, you can simulate them. It also enables you to program with blocks. Rather than having to write it from scratch, you may download/copy-paste the produced code into Arduino IDE later to program the real Arduino board.

We used jumper wires and a beard board to connect all of the devices to the Arduino Uno as shown in Fig. 10. Pin no A1, 10, 9, 6, 5, 3 of Arduino Uno connected with LCD. Arduino pin A0 is used to read the temperature sensor output. Arduino pin no 7 is used to read the ultrasonic sensor output which is constant means it would not change. Here, we also define pin no 8 for servo. Here some of the pins are behave either as an input or an output. Here pin no 2 and A0 behave as input and pin no 4, 11, 12, 13 are behave as output. Algorithm for smart home automation system.

The ultrasonic distance sensor is put on the top of the door in a smart home system to measure the distance when a person approaches the room. When the distance is less than 40 cm, the dc motor opens the door, waits 2 s, and then closes it. When a person enters the room, the PIR sensor detects motion, and the lights switch on as a result. Otherwise, the sensor will continue to detect motion. Lines 16–20 illustrate how a fan that operates at ambient temperature works. The fan is turned off if the temperature is less than or equal to 24 °C; on the other hand, if the temperature is greater than or equal to 24 °C, the fan is turned on, and the speed of the fan is proportional to the temperature. As a result, as the temperature rises, the fan's speed rises with it.

When a person approaches the room in a smart home system, an ultrasonic distance sensor is mounted on the top of the door to monitor the distance. When the distance between the door and the threshold is less than 40 cm, the DC motor opens the door, waits 2 s, and then closes it (1–6). The lights went on when a person entered the room and the PIR sensor detected motion. If not, the sensor will continue to detect movement (7–11). Lines 12–18 illustrate how a fan that operates at ambient temperature works. If the temperature is less than or equal to 24 °C, the fan is turned off; if the temperature is greater than 24 °C, the fan is switched on, and the fan's speed is proportionate to the temperature. As a result, the fan's speed increases as the temperature rises. Method 1 shows how the algorithm works in detail.

Algorithm 1: Smart Home System Algorithm

```
distance sensed by the Ultrasonic Distance sensor
if distance less than 40 cm then
DC motor: open the door
Else
Close the door and keep sensing
End if
If PIR sensor detects the motion, then
Turn on the light (LED)
Else
Keep sensing
End if
Temperature sense by the LM35 Temperature Sensor
If the temperature is below or equal to 24 °C,
Fan was switched off.
else
If the temperature rises above 24 °C, the fan is turned on.
end if
end if
```

This study is divided into three sections: sensing, monitoring, and control. Sensors such as ultrasonic sensors, PIR sensors, and temperature sensors are used for sensing, while our microcontroller unit, the Arduino UNO, is used for controlling and monitoring. Arduino UNO is used to connect the sensors and appliances. The value of sensors has a significant impact on the state of our equipment. UDS measures the distance between the door and the servomotor opens and closes the door accordingly. The PIR sensor detects human or animal movement, whether or not there is LED lighting present. The temperature sensor detects the temperature and controls whether the fan is turned on or off using a Dc motor. The status of our devices is uploaded to the cloud, and the client can view it on his PC as well as his mobile phone. The Arduino UNO controls the devices based on the value provided by sensors.

6 Conclusion

In this paper, predominant attention is about controlling and operating various smart home appliances remotely. This domestic automation approach provide more performance in utilization of electricity. And, it makes domestic as a smart location to live on, since Arduino Uno board could be very beneficial and desired which makes the IoT structures value powerful with required ultra-low power consumption functionality. The home computerization framework of the future will be more brilliant, faster, and easier to scale. A lot of examination and work is being done in this likely to coordinate artificial knowledge, which will fundamentally affect this field and may regard the accomplishment of a totally talented sharp home system.

References

1. Mihalache A (2017) Wireless home automation system using IoT. Informatica Economica 21(2)
2. Satapathy, Lalit Mohan, Samir Kumar Bastia, and Nihar Mohanty. "Arduino based home automation using Internet of things (IoT).“ International Journal of Pure and Applied Mathematics 118.17 (2018): 769–778.
3. Tejesh BSS, Neeraja S (2018) A Smart Home Automation system using IoT and Open Source Hardware. International Journal of Engineering & Technology 7(27):428
4. Singh, Shweta, and Kishore Kumar Ray. "Home automation system using internet of things.“ International Journal of Computer Engineering and Applications (2017): 1–9.
5. Pavithra, D., and Ranjith Balakrishnan. "IoT based monitoring and control system for home automation.“ 2015 global conference on communication technologies (GCCT). IEEE, 2015.
6. Abdulraheem, Ahmad Sinali, et al. "Home automation system based on IoT.“ (2020).
7. Singh, Harsh Kumar, et al. "A step towards Home Automation using IOT.“ 2019 Twelfth International Conference on Contemporary Computing (IC3). IEEE, 2019.
8. Kodali, Ravi Kishore, and Subbachary Yerroju. "Energy efficient home automation using IoT.“ 2018 International Conference on Communication, Computing and Internet of Things (IC3IoT). IEEE, 2018.
9. Nitu, Adiba Mahjabin, Md Jahid Hasan, and Md Shahin Alom. "Wireless home automation system using iot and paas.“ 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT). IEEE, 2019.
10. Mandula, Kumar, et al. "Mobile based home automation using Internet of Things (IoT).“ 2015 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT). IEEE, 2015.
11. Somani, Shradha, et al. "IoT based smart security and home automation.“ 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA). IEEE, 2018.
12. Sivapriyan, R., K. Manisha Rao, and M. Harijyothi. "Literature review of iot based home automation system.“ 2020 Fourth International Conference on Inventive Systems and Control (ICISC). IEEE, 2020.
13. Mahalakshmi G, Vigneshwaran M (2017) IOT based home automation using Arduino. Int. J. Eng. Adv. Res. Technol 3(8):1–6
14. Venkatesh K et al (2018) IoT based home automation using raspberry Pi. Journal of adv research in dynamical & control systems 10(7):1721–1728
15. Sehgal T, More S (2017) Home automation using IoT and mobile app. International Research Journal of Engineering and Technology (IRJET) 4(02):694–698
16. Shinde, H. B., et al. "Smart home automation system using android application.“ System 4.04 (2017).
17. Amudha A (2017) Home automation using IOT. International Journal of Electronics Engineering Research 9(6):939–944
18. Sharma, M. L., Sachin Kumar, and Nipun Mehta. "Smart home system using IoT.“ International Research Journal of Engineering and Technology 4.11 (2017): 1108–1112.
19. Wadhwani, Siddharth, et al. "Smart home automation and security system using Arduino and IOT.“ International Research Journal of Engineering and Technology (IRJET) 5.2 (2018): 1357–1359.

20. Jabbar, Waheb A., et al. “Design and implementation of IoT-based automation system for smart home.“ 2018 International Symposium on Networks, Computers and Communications (ISNCC). IEEE, 2018.
21. Singh, Himanshu, et al. “IoT based smart home automation system using sensor node.“ 2018 4th International Conference on Recent Advances in Information Technology (RAIT). IEEE, 2018.
22. Patchava, Vamsikrishna, Hari Babu Kandala, and P. Ravi Babu. “A smart home automation technique with raspberry pi using iot.“ 2015 International conference on smart sensors and systems (IC-SSS). IEEE, 2015.
23. Tseng, Chwan-Lu, et al. “An IoT-Based Home Automation System Using Wi-Fi Wireless Sensor Networks.“ 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC). IEEE, 2018.
24. Dey, Shopan, Ayon Roy, and Sandip Das. “Home automation using Internet of Thing.“ 2016 IEEE 7th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON). IEEE, 2016.
25. IoT Simulator [online]. Available: https://www.tinkercad.com.

Brain Depletion Recognition Through Iot Sensors Empowered with Computational Intelligence

Shaji. K. A. Theodore, K. Selvakumar, and G. Revathy

Abstract The world has been inspired by the development of Internet of Things enabling applications in recent years, which have brought outstanding solutions to a number of problems. This fast-growing industry is led by wireless sensor networks, radio frequency identification, and smart mobile technologies. Smart medical devices and wearables, for example, play an important part in the Internet of Things, as they may collect a variety of longitudinal patient-generated health data while also presenting preliminary diagnosis options. As part of their efforts to support patients utilizing IoT-based solutions, experts exploit the capabilities of machine learning algorithms to provide effective solutions in bleeding detection. This chapter outlines a smart IoT-based system for human brain haemorrhage diagnostics that employs deep learning algorithms to lower fatality rates and provide accurate treatment suggestions. To classify the pictures from the computed tomography scans for the intracranial dataset, the Nave Bayes Classification and Recurrent Neural Network were utilized. The classification results for the Naive Bayes classification and Recurrent neural network are superior to prior techniques such as support vector machines, knn, and kmediods. According to the findings, the recurrent neural network outperforms at classifying intracranial imagery. The output of the classification tool includes information on the type of brain haemorrhage, which aids in the validation of an expert's diagnosis and is utilised as a learning tool for trainee radiologists to eliminate errors in current systems.

Shaji. K. A. Theodore
Faculty of IT–Networking, Department of IT, University of Technology and Applied Sciences, Al-Masnaah, Sultanate of Oman
e-mail: shaji@act.edu.om

K. Selvakumar
Department of IT, Faculty of Engineering and Technology, Annamalai University, Chidmabaram, India

G. Revathy (✉)
School of Computing, SASTRA Deemed to Be University, Thanjavur, India
e-mail: revathyjayabaskar@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_20

Keywords IoT · Wireless sensor networks · Machine learning algorithms · Naïve bayes classification · Recurrent neural networks

1 Introduction

The Internet of Things (IoT) is a network of physical objects or people known as “things” that are connected with software, electronics, networks, and sensors in order to collect and exchange data. The Internet of Things (IoT) aims to extend internet connectivity beyond traditional devices like computers, smartphones, and tablets to everyday goods like toasters. As a result of the Internet of Things, which harnesses the power of data collection, AI algorithms, and networks to improve aspects of our life, virtually everything becomes “smart.” The Internet of Things includes things like a person with a diabetes monitor implant, an animal with tracking gadgets, and so on.

Based on Bayes’ theorem and strong (naive) independence assumptions across features, Naive Bayes classifiers are a sort of “probabilistic classifier” (see Bayes classifier). They are one of the most basic Bayesian network models, however when paired with kernel density estimation, they can achieve better accuracy levels.

In a learning task, the number of parameters required by Naive Bayes classifiers is proportional to the number of variables (features/predictors). In contrast to many other types of classifiers, maximum-likelihood training can be accomplished by evaluating a closed-form expression in linear time rather than by iterative approximation, which is time-consuming. Naive Bayes models are referred to as simple Bayes or independent Bayes in statistics and computer science.

A stroke is a type of brain haemorrhage. When a cerebral artery ruptures, localised bleeding occurs in the surrounding tissues. The bleeding causes the brain cells to die. The Greek term for blood is hemo. The exact definition of haemorrhage is “blood bursting forth.” Brain haemorrhages are referred to as cerebral haemorrhages, intracranial haemorrhages, and intracerebral haemorrhages. They account for around a third of all strokes. When blood from trauma irritates brain tissues, swelling occurs. The medical word for this condition is cerebral edoema.

A hematoma is a mass formed by blood that has gathered together. These conditions put more pressure on nearby brain tissue, reducing vital blood flow and causing brain cells to die. Bleeding can occur inside the brain, between the brain and its surrounding membranes, between the layers of the brain’s covering, or between the skull and the brain’s covering. A head injury occurs when there is trauma to the scalp, skull, or brain. When the brain is injured, the result is a traumatic brain injury, or TBI.

An intracranial hematoma (ICH) is a clot in the brain that forms beneath the skull. Hematomas in the brain are categorised according to where they originate and range in severity from mild to severe. This chapter explains how IoT sensors and deep learning algorithms are being utilised to predict ICH in humans and save lives. The high pandemic scenario that occurred in Covid’19 resulted in many deaths, and many people are scared to go to the hospital for fear of acquiring one of these diseases. Based on Naive Bayes Classification and Recurrent Neural Networks, this article provides an IoT-based model for brain haemorrhage diagnosis (RNN).

2 Naïve Bayes

The Naive Bayes model is easy to build and works well with large datasets. Because of its simplicity, Naive Bayes is said to outperform even the most sophisticated classification algorithms. Using *P*(*c*), *P*(*x*), and *P*(*x* |*c*) and the Bayes theorem, you can compute the posterior probability *P*(*c* |*x*) from *P*(*c*), *P*(*x*), and *P*(*x* |*c*). Consider this formula:

$$\underset{\text{Posterior Probability}}{P(c\mid x)} = \frac{\overset{\text{Likelihood}}{P(x\mid c)}\,\overset{\text{Class Prior Probability}}{P(c)}}{\underset{\text{Predictor Prior Probability}}{P(x)}}$$

$$P(c\mid X) = P(x_1\mid c)\times P(x_2\mid c)\times\cdots\times P(x_n\mid c)\times P(c)$$

Above,

- *P*(*c* |*x*) is the posterior probability of class (*c*, target) given predictor (*x*, attributes).
- *P*(*c*) is the prior probability of class.
- *P*(*x* |*c*) is the likelihood which is the probability of predictor given class.
- *P*(*x*) is the prior probability of predictor.

3 Recurrent Neural Networsks

A recurrent neural network (RNN) is an artificial neural network with nodes connected in a directed graph that follows a temporal sequence. This allows it to act in a time-dependent manner. By utilising their internal state, RNNs, which are built from feedforward neural networks, can process variable length sequences of inputs (memory). As a result, tasks such as unsegmented, connected handwriting recognition and speech recognition are now available.

4 Dataset

PSG polysomnography measurements (ECG & EEG), blood pressure, and airflow are all recorded together with the date and time. When blood flow drops to 25–35 ml/100 g/min, alpha frequencies disappear, while theta frequencies arise when blood flow drops to 17–18 ml/100 g/min, according to a study. When blood flow is between 12 and 18 ml/100 g/min, neurons twitch, causing transmembrane gradients to drop and brain cell death. EEG activity becomes quieter and cellular damage is irreversible when blood flow is less than 10–12 ml/100 g/min. It is possible to make the same prognosis even if the patient is sleeping. The numbers are meticulously recorded.

5 Proposed Work System

The measurements of blood flow, blood pressure, and air flow are connected to an Arduino board with a Wifi module and fed to a Naive Bayes Classifier. The Naive Bayes Algorithm classifies the values passed by the IOT sensor into two categories: safe and non-safe. Intracranial pictures are captured and sent to recurrent neural networks in individuals who are in the non-safe zone. Recurrent neural networks are utilised to anticipate patients who are at risk of haemorrhage, and the message is instantly sent on to the doctor or the patient's family, and the necessary actions are taken. The situations of patients with non-hemorrhage are frequently evaluated, and values are examined on a regular basis.

6 Block Diagram

See Fig. 1.

7 Implementation

The implementation of IOT devices are done with Arduino board connected with GSM module. The circuit connection with sample sensors is given below. The sensors are connected and a sample output temporarily recorded using the sensors (Fig. 2).

The output is passed to Naïve Bayes and the execution is performed using Orange Tool (Fig. 3).

The sensor values are taken in an Excel file and Naïve Bayes Classifier learning model is applied and the values are tested (Fig. 4).

Naïve Bayes shows 98% accuracy result (Fig. 5).

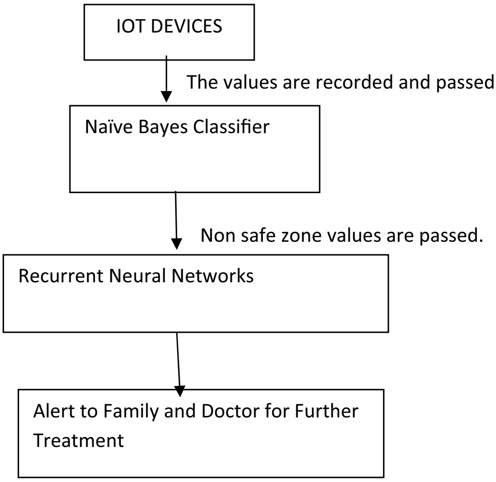

IOT DEVICES
The values are recorded and passed
Naïve Bayes Classifier
Non safe zone values are passed.
Recurrent Neural Networks
Alert to Family and Doctor for Further Treatment

Fig. 1 Block diagram of the proposed system

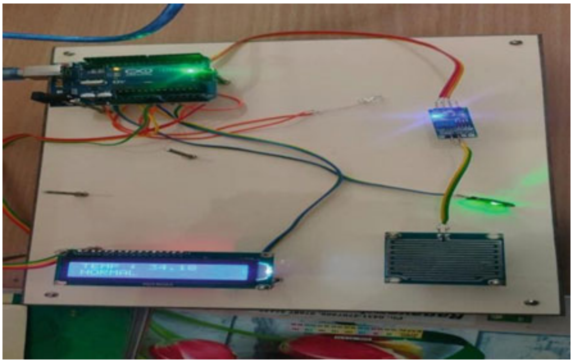


Fig. 2 Circuit diagram

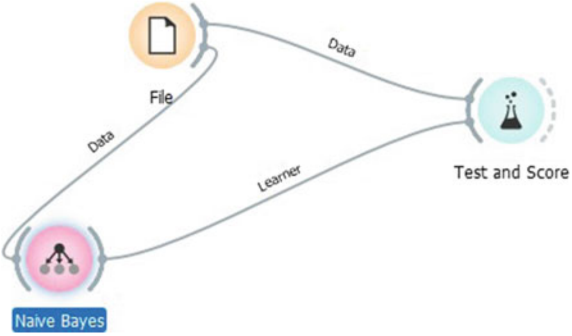
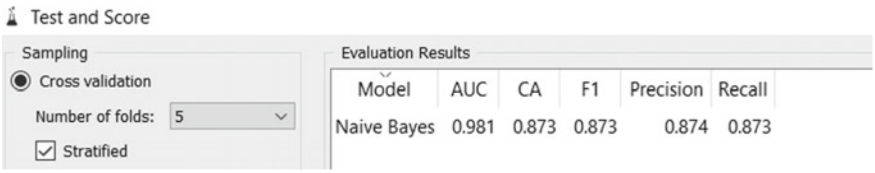


Fig. 3 Connections of Naïve Bayes for prediction



Test and Score

Sampling

● Cross validation

Number of folds: 5

☑ Stratified

Evaluation Results

Model	AUC	CA	F1	Precision	Recall
Naive Bayes	0.981	0.873	0.873	0.874	0.873

Fig. 4 Test result of Naïve Bayes

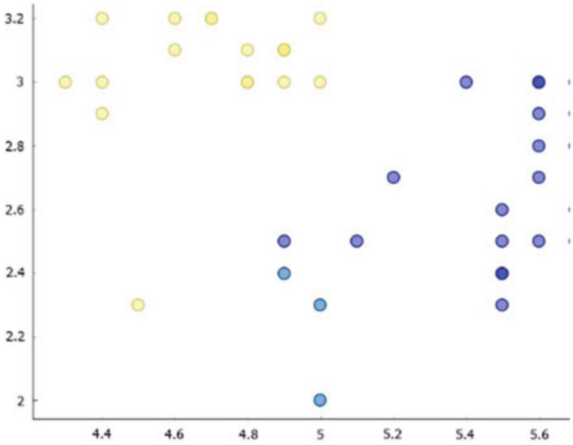


Fig. 5 The values are classified into safe zone and Non-safe zone

Naïve bayes split the complete data set into two categories one is safe zone data set which is indicated blue in color and another one is non-safe zone data set which is indicted yellow in color. The values that contain non-safe zones are verified for intracranial images and the same are passed to Recurrent neural networks for further investigation. For the other values the process is continuously repeated (Fig. 6).

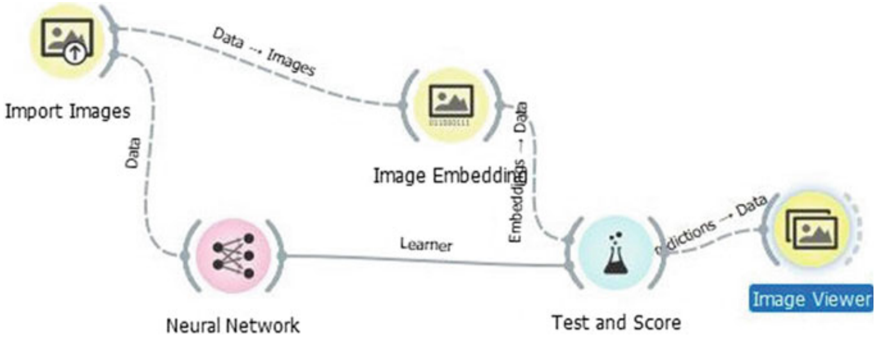


Fig. 6 Neural network implication with intracranial images

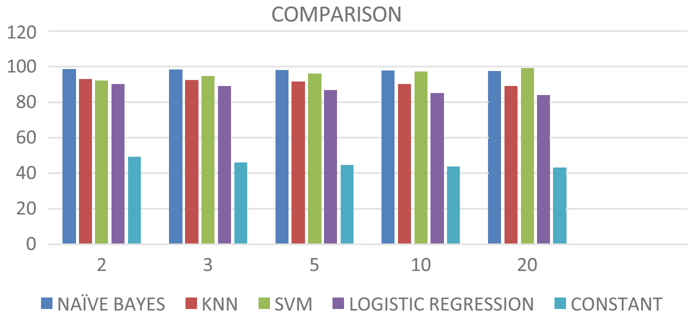


Fig. 7 Naïve Bayes and RNN comparison with all other algorithms

Recurrent Neural networks predicts the hemorrhage patient and through Wifi module Arduino board it informs to the Patient family and Doctor. The overall accuracy of Recurrent neural networks is 97% (Fig. 7).

Naïve Bayes combined with RNN shows higher values when compared each time with all other machine learning algorithms. Hence the result will be more accurate than any other algorithms.

8 Conclusion

During a pandemic situation like Covid 19 it's really hard for admitting every patient in a hospital and test the results for them. Hence by using our proposed method more than many a lives are saved and the patients who are in normal condition need not visit any hospital and get acquainted to new diseases. The patient who is in need of treatment or the patient who is in emergency alone can be admitted in hospital and treatment can be done. Moreover the total set up of components are very less, it's around 350 rs. so they can easily adopt to this system and need not waste much in MRI and CT scans often, once needed alone they can go with further treatments. Our system gives an accurate of 98% result, so a patient life will be surely saved.

References

1. Chuang H-C, Shih C-Y, Chou C-H, et al (2009) The development of a blood leakage monitoring system for the applications in hemodialysis therapy
2. Lin C-H, Chen W-L, Li C-M, et al (2005) Assistive technology using integrated flexible sensor and virtual alarm unit for blood leakage detection during dialysis therapy
3. Tekale S, Shingavi P, Wandhekar S, Chatorikar A, Kamalnayan VP (2015) Prediction of chronic kidney disease using machine learning algorithm

4. Revathy G, Kavitha NS, Senthivadivu K, Sathya D, Logeshwari P (Jan 2020) Girl child safety using IoT sensors and Tabu search optimization. Int J Recent Technol Eng (IJRTE) 8. ISSN: 2277-3878
5. Revathy G, Saravanan G, Madonna Arieth R, Vengateshwaran M (Nov 2019) Magnify Qos with Tabu & link scheduling in Wmn. Int J Recent Technol Eng (IJRTE) 8(4). ISSN: 2277-3878
6. Revathy G (Sep 2018) Mounting eminence of services in wireless mesh networks. Int J Res Anal Rev (ISSN 2349 5138)
7. Revathy G, Selvakumar K, Sustain route by tabu and amplified qos by distributed scheduling in wmn. Int J Recent Trends Eng Res (ISSN: 0973-7391)
8. Revathy G, Selvakumar K, Channel assignment using tabu search in wireless mesh networks. Wireless Pers Commun ISSN NO 09296212
9. Revathy G, Selvakumar K (March 2018) Increasing quality of services in wireless mesh networks. Int J Adv Res Comput Eng Technol 7(3). ISSN 22781323
10. Revathy G, Selvakumar K (Jan 2018) Escalating quality of services with channel assignment and traffic scheduling in wireless mesh networks. Cluster Comput. ISSN no 13867857
11. Revathy G, Selvakumar K (2017) Route maintenance using tabu search and priority scheduling in wireless mesh networks. J Adv Res Dyn Control Syst 9:sp–6. ISSN 1943023X
12. Zhang Z, Ho KM, Hong Y (2019) Machine learning for the prediction of volume responsiveness in patients with oliguric acute kidney injury in critical care. Critical Care
13. Tomasěv N, Glorot X, Rae JW, Zielinski M, Askham H, Saraiva A, et al (2019) A clinically applicable approach to continuous prediction of future acute kidney injury. Nature
14. Thakur N, Han CY (2021) A study of fall detection in assisted living: identifying and improving the optimal machine learning method. J Sens Actuator Netw
15. https://dias.library.tuc.gr
16. https://orange3.readthedocs.io

Auction-Based Deadline and Priority-Enabled Resource Allocation in Fog–IoT Architecture

Nikita Joshi and Sanjay Srivastava

Abstract Internet of Things (IoT) applications such as health care generate a large amount of critical data to be analyzed and acted upon in real time. Fog computing architecture has been proposed to handle this class of low-latency applications. Ownership of cloud and fog resources by different entities makes IoT applications scalable. Task allocation in such non-monolithic architecture while satisfying stringent IoT requirements such as deadline and priority with the monetary cost is a complex problem. In this paper, we propose a multi-attribute double auction-based task allocation algorithm. The objective of this algorithm is to maximize the utility of each stakeholder in the architecture such that it satisfies deadline and priority constraints. Deadline and priority-based double auction (DPDA) algorithm shows 41.07% improvement in system utility than double auction-based allocation and 4.8% more than the primary deadline and priority-based algorithm.

Keywords Fog–IoT architecture · Health care · Resource allocation · Auction mechanism · Double auction

1 Introduction

The population of the world is increasing day by day. It is expected to reach 9.8 billion by 2050 [1]. The elderly population is expected to raise 16% more than the total population between 2025 and 2050 [2]. Therefore, there will be increase in the need for continuous monitoring of elderly patients with chronic diseases. As predicted by the World Health Organization, there will be a shortage of 12.9 million healthcare workforce worldwide by 2035 [3]. IoT-based healthcare services are proposed to

N. Joshi (✉) · S. Srivastava
Dhirubhai Ambani Institute of Information and Communication Technology, Gandhinagar, Gujarat 382007, India
e-mail: nikita_joshi@daiict.ac.in

S. Srivastava
e-mail: sanjay_srivastava@daiict.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_21

handle this situation in which the patient is continuously monitored using sensors attached to his body and environment. These IoT-based service providers have collaborated with clinicians and technologists. They acquire the data from the patients, process it and then send notifications to hospitals or patients and caregivers [4].

Healthcare IoT applications generate a massive amount of data [5]. Most of the data are of no use if it is not processed in real time. IoT devices are resource constrained; making a real-time data processing challenging. There are existing cloud service providers (CSPs), including Amazon, Microsoft [6], and IBM [7], which have adequate resources for this IoT-based medical data processing tasks. Suh et al. [8] have proposed cloud IoT-based architectures for the patient monitoring system. Data collected from sensors are sent to a remote server known as a cloud node to process these architectures. Unfortunately, the delay between cloud and user is not acceptable by healthcare applications since they are critical and delay sensitive.

The fog computing paradigm is introduced to minimize the delay between users and the processing unit. Fog devices have fewer computing resources than the cloud, but the delay between fog devices and the user is lower than the cloud. Thus, fog is nearer to the user than the cloud in terms of delay. Healthcare applications such as fall detection system, epileptic seizure detection (EEG) monitoring [9, 10], electrocardiogram (ECG) monitoring [11], smart ambulance [12], and hypertension attack detection [13] are implemented using fog–IoT architecture.

Fog service providers (FSPs) are still under development. Many small fog service providers are not well known. Therefore, the user needs to contact the cloud service provider to get resources. Also, if a user is aware of fog service providers, they need to communicate with all FSPs and select one of them. This approach is time-consuming since each user communicates with service providers individually. CSP acts as a bridge between users and FSPs. Thus, a user needs resources; though CSPs have resources, they are far from the user, while FSPs have resources and are nearer to the user but can't directly contact the user. So, CSPs can get FSPs on rent, satisfy user requirements, and profit from the difference between user payment and FSPs.

2 Related Work

Recent studies [14–18] have worked on resource allocation in fog–IoT architecture with limited focus on some of the IoT requirements including deadline, priority, energy minimization, and delay minimization-based allocation. They considered monolithic architecture in which a single entity owned all the resources at a fog level and cloud level. Therefore, price is not considered while allocating the tasks.

In non-monolithic architecture, application provider rents resources from fog nodes or cloud nodes. Additionally, fog service providers and cloud service providers may not be the same. Non-monolithic architecture has several advantages. Firstly, scalability, as we are purchasing resources on rent, and many resource providers are available in the market. IoT application provider doesn't need to limit the number of users or the service's area.

Furthermore, better resource utilization was achieved; if each application provider implements resources for his requirements individually, when his users are not using resources, it gets wasted. Moreover, the IoT application provider doesn't need to focus on developing infrastructure for better management since it can be purchased on rent. They need to focus on application design only. Fourthly, Amazon, Microsoft, and Google are giant cloud providers for IoT applications. Amazon and Microsoft have also introduced fog computing facilities called AWS Greengrass and Microsoft IoT edge, respectively. Several studies [19–21] have emphasized only on money-based allocation in fog computing without considering IoT application requirements.

Furthermore, studies [22, 23] have focused on both; however, they can't satisfy IoT requirements. Firstly, the online nature of tasks means that IoT tasks are not known apriori. Secondly, real-time requirement means each task has a deadline associated with it. If the task can't be executed before that deadline, then it is not meaningful. Thirdly, each IoT task has some priority, and if resources are limited, a higher priority task must be performed first. In our previous work [24], Stackelberg game was used to allocate periodic as well as sporadic task and determine price for CSP. Limitation of this approach is Stackelberg game is leader follower game. CSP as a leader will decide price and quantity both. User's budget is not considered.

For non-monolithic architecture, deciding the price of service providers is also a big problem. Three types of pricing policies for cloud service providers are available in the market nowadays[25]: (i) static pricing in which all customers have to pay the same fixed price per unit of time. However, it has problems like over-provisioning, under-provisioning, and waste of resources. (ii) Dynamic pricing allows a provider to decide the price of the resources based on demand and supply of the resources in the market. However, it charges the same price to all customers, so users cannot have additional privileges by paying more. (iii) Forward auction-based pricing is implemented by amazon EC2. The user needs to bid the maximum price that the user is willing to pay for renting the unused computing instances. The highest bidder will get the resource. Auction helps in price discovery for fog and cloud service providers. Additionally, it is more suitable because of the perishable nature of resources.

Three types of auction mechanisms can be used for price discovery and allocation in fog–IoT architecture [25]: (i) forward auction in which users bid for resources and the highest bidders will get the resources; (ii) reverse auction in which sellers bid for selling their resources and customers will select the low-cost seller, and (iii) the double auction mechanism in which the user and seller send their bid and a third party (broker) will find matching among them. Regular auctions focus on price, which is not suitable for IoT applications with Quality of Service (QoS) requirements. In such a case, a multi-attribute-based auction mechanism should be used in which instead of only focusing on price, QoS requirements are also considered. For all these three kinds of auction mechanisms, multi-attribute-based variants are also applicable for resource allocation in fog–IoT architecture.

Agrawal et al. [26] have proposed reverse auction-based resource allocation in fog–IoT architecture, which also satisfies QoS requirements of IoT applications. The limitation of this work of a reverse auction mechanism is that only service providers can bid. Users cannot send bids. Thus, the budget of users is not considered during allocation. Peng et al. [27] have proposed double auction-based resource allocation, which creates a bipartite graph between users and fog nodes. The weight of edge in the graph is decided based on QoS requirements. After that, maximum matching of a bipartite graph is found to perform the allocation. Since they are using matching to find allocation, one fog can serve only one user though it has other resources left that can satisfy other users' requirements.

This paper is organized as follows. Initially, we have described system model in Sect. 3 followed by the formulation of the problem statement in Sect. 4. The method of task allocation algorithm is explained in Sect. 5. Finally, simulation results are discussed in Sect. 6 with conclusions.

3 System Model

We used three tier architecture as shown in Fig. 1 in which there are N IoT users, M FSPs, and one CSP. Each user is registered with a CSP. All FSPs are also registered with CSP. Whenever a user wants resources, a user sends a request to CSP. CSP finds the appropriate FSP or may decide to execute the task itself. The user sends input data to the allocated fog node or cloud and gets the results. The symbols used in this paper are listed in Table 1.

Fig. 1 System model

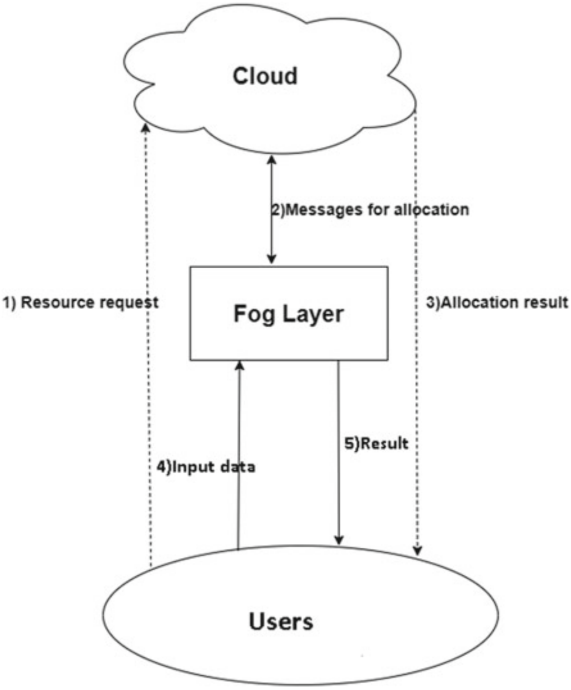


Table 1 Notations

Symbol	Definition
N	Total number of users
M	Total number of FSP
q_i	Resource requirement of ith user
p_i	Priority of ith user
b_i^u	Bid of ith user
b_j^f	Bid of jth fog
U_i^h	Utility of user i
U_j^f	Utility of FSP j
U^c	Utility of CSP
U_s	System utility
D_i^s	Soft deadline of user i
D_i^h	Hard deadline of user i
V_i	Valuation of task for ith user
C_i	Payment to be made to CSP by ith user
L_i	Penalty of CSP by ith user
R_i	Reward by ith user
r_j	Price of jth FSP for one unit of RI
A_j	Available resources at jth FSP
e_i	Energy required for processing of one RI at CSP
d_{ij}	Total delay to execute task of ith user on jth FSP
τ_{ij}	Decision variable of ith user and jth FN
E_j	Energy cost required to process one RI by jth FN
t_i	Service delay for user i

Resources are modeled as resource instance(RI). Each RI has a fixed configuration of resources such as memory size, CPU cycles, and network bandwidth. Each task has a resource requirement q_i which shows the number of RIs required. Each task has priority p_i associated with it. In the limited resources situation, high-priority tasks should be executed first. IoT applications are delay sensitive. Therefore, each task has a deadline (soft deadline D_i^s, hard deadline D_i^h) associated with it. After the deadline is over, its result is not useful. The user sends these task characteristics and his bid b_i^u to CSP based on which CSP allocates FSPs to them.

Ins non-monolithic architecture, each entity has its own utility functions. Users of cloud–fog services are healthcare service providers here. They get money from the patients for executing tasks. That is defined as the valuation of the task here. The utility of the user is the difference between the valuation of the task and the payment to be made to the cloud. Also, the valuation of a task for healthcare applications is decreased linearly after soft deadline [14].

$$U_i^h = V_i - C_i - \frac{t_i - D_i^s}{t_i} \tag{1}$$

here, V_i is valuation of task for ith user, and C_i is payment to be made by ith user to cloud.

$$C_i = (1 - L_i + R_i) * b_i^u \tag{2}$$

Here, L_i is penalty if cloud can't satisfy deadline requirement, and R_i is reward given by ith user. The penalty function is as following

$$L_i = \begin{cases} (\frac{t_i - D_i^s}{t_i})^{p_i}, & \text{if } D_i^h > t_i > D_i^s \\ 1, & t_i > D_i^h \\ 0, & \text{otherwise} \end{cases}$$

Here, t_i is a actual service delay. Reward is calculated based on priority of task. For processing higher priority tasks, cloud is given reward.

$$R_i = (1 - \frac{1}{p_i}) \tag{3}$$

CSPs receive payment from users and need to pay to FSPs. Also, if it executes tasks itself, energy cost should be considered. Therefore, the utility of the cloud is calculated as follows:

$$U^c = \sum_{i=1}^{N} q_i(C_i - \tau_{i,0}e_i - \sum_{j=1}^{M} \tau_{ij}r_j) \tag{4}$$

Here, r_j is price of one unit of RI for jth FSP. $\tau_{i,0}$ is 1 if task is allocated to cloud itself. e_i is energy cost to execute task of ith user at cloud.

Utility of FSP is as following:

$$U_j^f = \sum_{i=1}^{N} q_i\tau_{ij}(r_j - E_j) \tag{5}$$

E_j is energy cost to execute task by jth FSP

If we consider that CSP and FSPs are single entity, then total system utility U_s is defined by

$$U_s = \sum_{i=1}^{M} U_j^f + U_c \tag{6}$$

4 Problem Statement

The main goal of this research work is to design an algorithm which performs deadline and priority-based allocation with following objectives:

$$\text{Maximize} \qquad \sum_{i=1}^{N}\sum_{j=1}^{M} U_i^h, U_j^f, U_c$$

subject to

$$\begin{aligned} & t_i \leq D_i^h,\ i = 1, \ldots, N. \\ & U_i^h \geq 0,\ i = 1, \ldots, N. \\ & U_j^f \geq 0,\ j = 1, \ldots, M. \\ & U_c \geq 0 \end{aligned}$$

This kind of problem is called the joint optimization problem. It is difficult to maximize the utility of all three entities simultaneously [21]. For simplification, we have assumed that CSP and FSPs are owned by same entity, so we are trying to maximize system utility and user utility. We have proposed heuristic-based allocation algorithm which uses double auction mechanism to solve this problem.

5 Proposed Algorithm

We have assumed that total time is divided in rounds. At the beginning of each round, CSP receives requests from user. CSP checks if the request can be fulfilled before soft deadline D_i^s, then it executes task itsef else it runs the allocation algorithm. The algorithm is divided in two modules, namely mapping of users and FSPs and price determination. The mapping algorithm uses double auction mechanism which matches highest prioritized and highest paying user to lowest cost FSP which can also satisfy deadline requirement of task. Input of this Algorithm 1 is Q vector which is list requests from users which contains $(q_i, p_i, b_i^u, D_i^s, D_i^h)$ for all users, I vector which contains (A_j, b_j^f) for each fog nodes and delay matrix d which contains delay between each user and cloud as well as all fog nodes. It returns allocation matrix τ which shows mapping of users and FSPs.

This allocation matrix is sent to users. Users send their input data to allocated fog or cloud and get the result. After that users send the QoS report to CSP. QoS report which contains actual service delay t_i of executing task based on which CSP determines the payments of users and FSPs. This algorithm calculates bill for user C_i that shows amount that user has to pay to CSP and r_j which is used to calculate payment to be made to fog by CSP. To ensure truthfulness, we are using Vickrey auction [28] to determine r_j. Algorithm 2 shows the details of payment calculation.

Algorithm 1: Mapping of users and FSPs

```
Input: $Q, I, d$
Output: $\tau$
for every user do
    $\tau_{ij} = -1$
    $b_i^u = b_i^u / q_i$
Sort $Q$ based on $p_i$ and $b_i^u$ in descending order:
Sort $I$ based on $b_j^f$ in ascending order
for each $i \epsilon Q$ do
    for each $j \epsilon I$ do
        if $D_i^s \geq d_{ij}$ and $A_j \geq q_i$ then
            $\tau_{ij}$=1
            $A_j = A_j - q_i$
```

Algorithm 2: Price determination

```
Input: $Q, I, t, \tau$
Output: $C_i, r_j$
for every user $i$ do
    $C_i$=0
    if $t_i >= D_i^h$ then
        $L_i = 1$
    else if $D_i^h >= t_i >= D_i^s$ then
        $L_i = (\frac{t_i - D_i^s}{t_i})^{p_i}$
    else
        $\mathrm{L_i}$=0
    $R_i = (1 - \frac{1}{p_i})$
    $C_i = (1 - L_i + R_i) b_i^u$
Sort $I$ based on $b_j^f$ in ascending order
for each $j \epsilon I$ do
    for each $i \epsilon Q$ do
        if $\tau_{ij} = 1$ then
            $r_j = b_{j+1}^f$
```

6 Simulation and Results

We have implemented the algorithm in network simulator (NetSim) and Python, in which we have derived delay between users and FSPs or cloud from NetSim. Parameters of NetSim are shown in Table 2.

For each application, we have sent 100 packets out of which 50 of them are request for allocation packet and 50 are execution packet in alternate manner. Other parameters set in Python code are mentioned in Table 3.

We have compared DPDA algorithm with the following algorithms:

1. Double Auction-based allocation (DA-allocation): [25] has proposed double auction-based allocation in which users are sorted in descending order based on bid, and FSPs are sorted in ascending order based on the bid, and then, the first

Table 2 NetSim parameters for algorithm implementation

Parameter	Value
Number of sensors	10
Number of routers (fog nodes)	3
Number of wired node (cloud)	1
Packet size	1000 KB
Inter packet arrival time	1 s
Simulation time	100 s

Table 3 Python parameters for algorithm implementation

Parameter	Value
A_j	300
q_i	$A_j * \eta$
η	[0.5, 1.5]
p_i	1,2
D_i^s	Normal(750, 75) ms
b_i^u	Normal(1000, 2000)
b_j^f	Normal(200, 100)

user is mapped to first FSP. Thus, the highest paying user is matched with the low-cost FSP. This algorithm performs allocation without considering IoT QoS requirements

2. Deadline and priority-based allocation (DP-allocation): This is a basic IoT algorithm that sorts users based on priority. The highest priority user is assigned to the nearest fog node without considering the price difference of fog nodes.

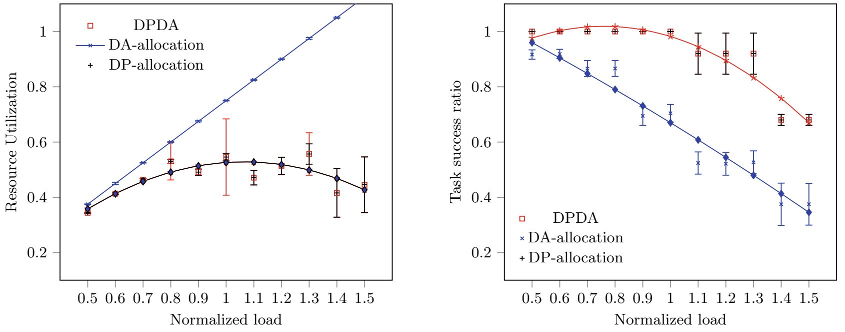
Following parameters are compared for these three algorithms:

1. Resource utilization (Ru_i): It is ratio of total resources allocated and total resources available

$$Ru_i = \sum_{i=1}^{N} \sum_{j=1}^{M} \frac{\tau_{ij} q_i}{A_j} \tag{7}$$

2. Task success ratio: It is ratio of task completed successfully before deadline and total task allocated by algorithm.
3. System utility: Sum of utility of cloud and fog calculated by Eq. 6.

As shown in Fig. 2a, resource utilization of DPDA and DP-allocated are the same because both of them perform allocation based on deadline and priority. Ru_i of DA-allocation is 37.07% more than DPDA since while doing allocation, it considers price only. So, some of the allocated tasks may not satisfy deadline requirements, but since they satisfy price criteria, they are allocated. As shown in Fig. 2b, the task



(a) Resource utilization vs. Normalized load

(b) Task success ratio vs. Normalized load

Fig. 2 Comparison of DPDA with DP-allocation and DA-allocation

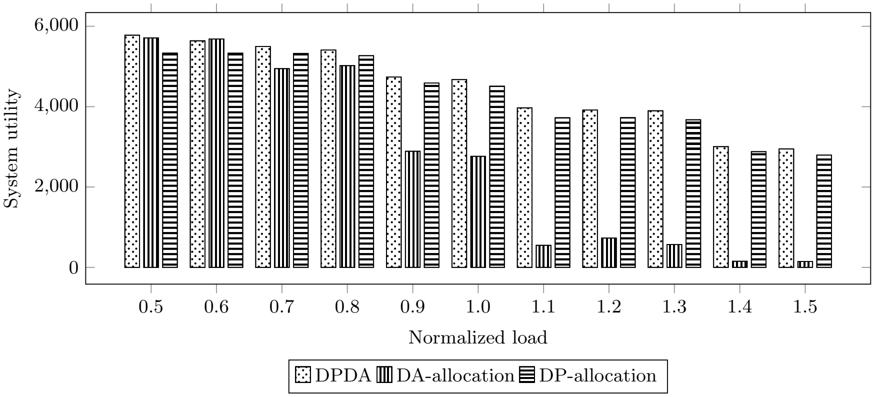


Fig. 3 System utility versus normalized load

success ratio of DPDA and DP-allocation is almost the same since they both accept tasks after satisfying the delay requirement. The task success ratio of DA-allocation is 27.97% less than DPDA since there are more tasks allocated, but many of them are not able to satisfy delay requirements.

As shown in Fig. 3 system utility for DPDA is highest. DA-allocation has 41.07% lower utility than DPDA since it allocates more resources, but many of them can’t satisfy deadline requirements. Therefore, execution cost and the penalty is higher than the revenue, which decreases the utility of the system. DP-allocation has 4.8% lower utility than DPDA since while allocating it has just considered the nearest fog node, not the low-cost fog node.

7 Conclusion

We have proposed a deadline and priority-based double auction mechanism which performs the allocation of tasks received from healthcare service providers. DPDA focuses on price as well as deadline and priority. DPDA is compared with DA-allocation and DP-allocation. Results show that DA-allocation allocates 37.07% more resources but is unable to complete all those tasks before the deadline, leading to lowering down system utility by 41.07% since it wasters execution cost plus penalty is charged by user. DP-allocation allocates almost the similar amount of tasks as DPDA, but without considering the price during allocation, therefore, it has lower system utility by 4.8% than DPDA. In the future, we envisage to involve sporadic tasks in to update this algorithm.

References

1. Population Reference Bureau (2017) https://www.prb.org/2018-world-population-data-sheet-with-focus-on-changing-age-structures/(2019), [Online; Accessed 10 Apr 2019]
2. An Aging World 2015 (2019) https://www.census.gov/library/publications/2016/demo/P95-16-1.html. [Online; Accessed 10 Apr 2019]
3. Global Health Workforce Shortage to Reach 12.9 Million in ComingDecades (2017) https://www.who.int/mediacentre/news/releases/2013/health-workforce-shortage/en/[Online; Accessed 10 Apr 2019]
4. Verma P, Sood SK (2018) Fog assisted-iot enabled patient health monitoring in smart homes. IEEE IoT J
5. Number of Connected Things/Devices WorldWide byVertical From 2015 to 2021. https://www.statista.com/statistics/626256/connected-things-devices-worldwide-by-vertical/(2019), [Online; accessed 10-4-2019]
6. Microsoft Azure (2019) https://azure.microsoft.com/en-in/[Online; Accessed 10 Apr 2019]
7. IBM cloud (2019). https://www.ibm.com/internet-of-things[Online; Accessed 10 Apr 2019]
8. Suh Mk, Chen CA, Woodbridge J, Tu MK, Kim JI, Nahapetian A, Evangelista LS, Sarrafzadeh M (2011) A remote patient monitoring system for congestive heart failure. J Med Syst 35(5):1165–1179
9. Cao Y, Chen S, Hou P, Brown D (2015) Fast: a fog computing assisted distributed analytics system to monitor fall for stroke mitigation. In: 2015 IEEE international conference on networking, architecture and storage (NAS). IEEE, pp 2–11
10. Craciunescu R, Mihovska A, Mihaylov M, Kyriazakos S, Prasad R, Halunga S (2015) Implementation of fog computing for reliable e-health applications. In: 2015 49th Asilomar conference on signals, systems and computers. IEEE, pp 459–463
11. Akrivopoulos O, Chatzigiannakis I, Tselios C, Antoniou A (2017) On the deployment of healthcare applications over fog computing infrastructure. In: 2017 IEEE 41st annual computer software and applications conference (COMPSAC), vol 2. IEEE, pp 288–293
12. Oladimeji EA, Chung L, Jung HT, Kim J (2011) Managing security and privacy in ubiquitous ehealth information interchange. In: Proceedings of the 5th international conference on ubiquitous information management and communication. ACM, p 26
13. Sood SK, Mahajan I (2018) Iot-fog based healthcare framework to identify and control hypertension attack. IEEE IoT J
14. Jiang Y, Tsang DH (2018) Delay-aware task offloading in shared fog networks. IEEE IoT J 5(6):4945–4956

15. Sahni Y, Cao J, Yang L (2018) Data-aware task allocation for achieving low latency in collaborative edge computing. IEEE IoT J 6(2):3512–3524
16. Shah-Mansouri H, Wong VW (2018) Hierarchical fog-cloud computing for iot systems: a computation offloading game. IEEE IoT J 5(4):3246–3257
17. Shi Y, Chen S, Xu X (2017) Maga: a mobility-aware computation offloading decision for distributed mobile cloud computing. IEEE IoT J 5(1):164–174
18. Yang Y, Wang K, Zhang G, Chen X, Luo X, Zhou MT (2018) Meets: maximal energy efficient task scheduling in homogeneous fog networks. IEEE IoT J 5(5):4076–4087
19. Pham XQ, Nguyen TD, Nguyen V, Huh EN (2019) Joint node selection and resource allocation for task offloading in scalable vehicle-assisted multi-access edge computing. Symmetry 11(1):58
20. Xu J, Palanisamy B, Ludwig H, Wang Q (2017) Zenith: utility-aware resource allocation for edge computing. In: 2017 IEEE international conference on edge computing (EDGE). IEEE, pp 47–54
21. Zhang H, Xiao Y, Bu S, Niyato D, Yu FR, Han Z (2017) Computing resource allocation in three-tier iot fog networks: a joint optimization approach combining stackelberg game and matching. IEEE IoT J 4(5):1204–1215
22. Jin AL, Song W, Wang P, Niyato D, Ju P (2015) Auction mechanisms toward efficient resource sharing for cloudlets in mobile cloud computing. IEEE Trans Services Comput 9(6):895–909
23. Sun W, Liu J, Yue Y, Zhang H (2018) Double auction-based resource allocation for mobile edge computing in industrial internet of things. IEEE Trans Ind Inform 14(10):4692–4701
24. Joshi N, Srivastava S (2019) Task allocation in three tier fog iot architecture for patient monitoring system using stackelberg game and matching algorithm. In: 2019 IEEE international conference on advanced networks and telecommunications systems (ANTS). IEEE, pp 1–6
25. Baranwal G, Kumar D, Raza Z, Vidyarthi DP (2018) Auction based resource provisioning in cloud computing. Springer
26. Aggrawala A, Kumarb N, Vidyarthic DP, Buyyad R Multi-attribute combinatorial reverse auction model for resource procurement in fog integrated cloud architecture
27. Peng X, Ota K, Dong M (2020) Multiattribute-based double auction toward resource allocation in vehicular fog computing. IEEE IoT J 7(4):3094–3103
28. Ausubel LM et al (1999) A generalized vickrey auction. Econometrica

Research of Basic Characteristics of Wireless Sensor Networks for Energy Monitoring System

Ilkhom Siddikov, Doston Khasanov, Halim Khujamatov, and Ernazar Reypnazarov

Abstract One of the key indicators of monitoring systems of solar power supply sources is the design and organization of the communication architecture of remote data transmission. This paper presents wireless network technologies that can be used in the remote monitoring of telecommunications systems for solar power supply sources. Various research papers on these network technologies have been studied and compared. Comparisons are evaluated on important criteria of network technologies (throughput, power consumption, network size, etc.) and the network technology with the best performance was selected for the monitoring system of solar power supply sources.

Keywords Wireless Sensor Networks · Monitoring system · Solar power supply · Wireless network technologies · WiMAX · Wi-Fi · Bluetooth · ZigBee

1 Introduction

A wireless sensor network (WSN) is a network created by a large number of sensor nodes, in which each node is equipped with sensors designed to detect light, vibration, temperature, pressure, and similar physical events. The data received by these sensor nodes is transmitted to the receiving node through several intermediate nodes. This node can use the data locally or transmit it to an external network (such as the Internet). Multi-hop nodes allow the transmission of data from the source to the end node and extend the service coverage of the network. The data collected at the end node is transmitted to the computer for processing and visualized for the user via a communication channel [1, 2]. The sensor node hardware typically consists of four parts: a power and power control module, sensors, a microcontroller, and a wireless transmitter module [3–5].

I. Siddikov · D. Khasanov (✉) · H. Khujamatov · E. Reypnazarov
Tashkent University of Information Technology Named After Muhammed Al-Khwarizmi,
Tashkent, Uzbekistan 100200
e-mail: dhasanov0992@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_22

WSN can be placed in a special mode, unlike traditional sensor networks, which are clearly planned and placed in a predetermined position. Based on the general characteristics, WSN must meet the following requirements: expansion relative to the number of nodes in the network, self-organization, self-fixing, energy-saving, sufficient level of connection between nodes, simplicity, low cost, and small size of nodes [6].

However, it is important to consider the advantages and disadvantages of WSN. The advantages of WSN include: network configuration is carried out without the established infrastructure; suitability for hard-to-reach places (e.g. rivers, mountains, deserts, etc.); flexibility; requires a low-wire; possibility to install new devices at any time; central control and management.

Their disadvantages include: access point security is insufficient because all data is obtained from this point; low speed compared to wired network; the complexity of the configuration relative to the wired network; environmentally sensitive (walls, other wireless signals, barriers, etc.); the possibility of attacking the transmitted signal; and high hardware costs.

WSN applications include surveillance, monitoring and control and are mainly used for residential monitoring, facility monitoring, nuclear reactor control, fire detection, traffic monitoring, and power supply monitoring [7, 8]. The paper explored various network technologies used in WSN applications such as Worldwide Interoperability for Microwave Access (WiMAX), Wireless Fidelity Wi-Fi, Bluetooth, and ZigBee.

2 Problem Statement

An important task of monitoring systems is to remotely detect appeared problems in controlled systems and eliminate them quickly. The purpose of using energy monitoring system is to monitor the performance of the system and to develop solutions to increase its efficiency based on the collected reports. Typically, this system consists of 3 main components: the main control panel of the system (1), monitoring and automatic detection of access from request points for the monitored object (using current and voltage, temperature, lighting, and other similar sensors) (2), and activation of output components based on them (management, database formation, and transmission of monitoring data over the Internet) (3) [9, 10].

The main purpose of the monitoring system of solar power supply systems is to monitor the parameters installed at each facility using appropriate sensors, digitize the signals of the monitored parameters and transmit them serially to the server of the monitoring center via telecommunication channels (radio or wired channel) [11, 12]. One of the main challenges in designing a monitoring system is the establishment of a data transmission network on the scale of a solar power station (SPS) [13]. The main problems are the analysis and selection of wireless network technologies for the identification and monitoring of high-quality, reliable, and energy-efficient communication between sensor nodes [14]. Hence, the analysis of network technologies according to the above criteria, the study of research papers, and their comparative analysis constitute the solution to the problem.

3 Analyses Wireless Sensor Network Technologies

This section analyses wireless sensor networks and the wireless communication technologies used in them, such as WiMAX, Wi-Fi, Bluetooth, and ZigBee [15]. Their features, functions, advantages, and disadvantages were compared, and it was evaluated which of them is more suitable for remote monitoring of solar energy sources.

3.1 WSN Built on Wi-Max

WiMAX is a set of wireless broadband communication standards based on a set of IEEE 802.16 standards that provide multiple physical layers and media access control MAC address access [16]. WiMAX technology was developed as a wireless technology based on the standard of broadband connection of data to the end terminal as an alternative to wired network and DSL technologies [17, 18].

In the WSN of monitoring systems, WiMAX technology is usually used as a base station. That is, various sensors are located on the monitored object and they are connected to the microcontroller unit (MU). In this case, the object to be monitored must be coverage of WiMAX services [17, 19]. The general structure of WiMAX-based WSN is shown in Fig. 1. Each wireless sensor node consists of sensors, MU, and WiMAX wireless network module.

Today, the construction of WSN based on WiMAX technology for monitoring systems has not justified itself. Because the construction of such systems requires a large amount of resources and a lot of money [20, 21].

3.2 WSN Built on Wi-Fi

Wi-Fi is named IEEE 802.11 standard, which belongs to the family of wireless network technologies. This technology transmits data in the 2.4 GHz frequency range, which includes frequency division multiplexing technology. It is mainly used for fast data transmission over short distances as its transmission distance reaches 50–100 m [22]. Given such features of this technology, many scientists have used it as a wireless sensor network for monitoring systems.

Studies have shown that Wi-Fi technology can be used in the traditional way of collecting data, however as a solution for short-term transmission of large amounts of data over short distances. It has also been proven that the use of Wi-Fi technology

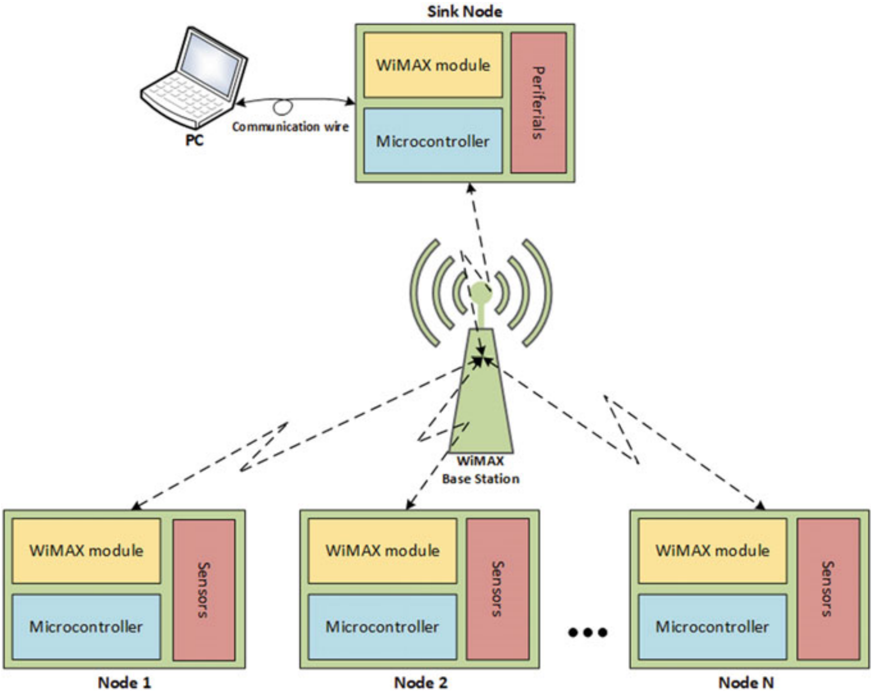


Fig. 1 WiMAX-based WSN for monitoring systems

as an intermediate signal booster is effective and can transmit data in a shorter time than ZigBee and bluetooth technologies [23].

For performance evaluated this network standard, the research work titled “Performance Evaluation of IEEE 802.11ah and its Restricted Access Window Mechanism” by Orod et al. was studied [24]. In this study, an analytical model for calculating the throughput and energy consumption of the IEEE802.11 standards was developed and the characteristics obtained. The analytical model includes certain collision and error probabilities and applies to both basic and RTS/CTS access mechanisms.

The study evaluated the throughput and energy consumption based on two main analytical equations by used (1) and unused (2) of RTS/CTS input mechanisms.

$$T_W^{\mathrm{RTS}} = \frac{p_{\mathrm{succ}}^{\mathrm{RTS}} \times 8 \times L_{pl}}{\sum_{n=0}^{r_{\mathrm{short}}-1} \sum_{m=0}^{r_{\mathrm{long}}-1} \left(\binom{n+m}{n} p_{\mathrm{col}}^{n} p_{\mathrm{err}}^{m} (1-p_{\mathrm{col}})^{m+1} (1-p_{\mathrm{err}}) t_{n,m} \right)}$$

$$\times \frac{1}{\begin{aligned} &\sum_{m=0}^{r_{\mathrm{long}}-1} \left(\binom{m+r_{\mathrm{short}}-1}{r_{\mathrm{short}}-1} p_{\mathrm{col}}^{r_{\mathrm{short}}} p_{\mathrm{err}}^{m} (1-p_{\mathrm{col}})^{j} t_{r_{\mathrm{short}},m} \right) \\ &+ \sum_{n=0}^{r_{\mathrm{short}}-1} \left(\binom{n+r_{\mathrm{long}}-1}{r_{\mathrm{long}}-1} p_{\mathrm{col}}^{n} p_{\mathrm{err}}^{r_{\mathrm{long}}} (1-p_{\mathrm{col}})^{r_{\mathrm{long}}} t_{n,r_{\mathrm{long}}} \right) \end{aligned}} \tag{1}$$

here:

T_W^{RTS} throughput when using RTS/CTS input mechanisms;
$p_{\text{succ}}^{\text{RTS}}$ the probability of successful transmission of the packet;
p_{col} the probability of a collision;
p_{err} the probability of transmitting packets with error;
L_{pl} payload;
r_{short} maximum number of retransmissions for RTS;
r_{long} maximum number of retransmissions for data packets;
n, m the number of collisions and errors, respectively.

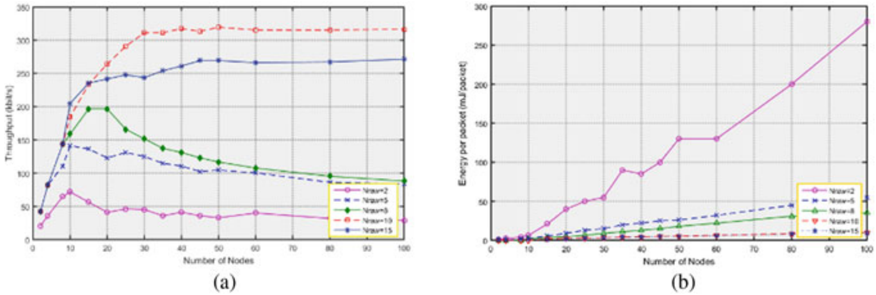
$$T_W^{\text{basic}} = \frac{p_{\text{succ}}^{\text{basic}} \times 8 \times L_{pl}}{\sum_{n=0}^{r_{\text{long}}-1} (p_{\text{fail}}^{n}(1 - p_{\text{fail}})t_n)} \times \frac{1}{p_{\text{fail}}^{r_{\text{long}}} \times t^{r_{\text{long}}}} \quad (2)$$

here:

T_W^{basic} basic throughput;
$p_{\text{succ}}^{\text{basic}}$ the probability of successful transmission of the packet;
p_{fail} the probability of failure transmission of packets;
$p_{\text{fail}}^{r_{\text{long}}}$ the probability of failure transmission of packets when the maximum number of retransmissions is exceeded;
$t^{r_{\text{long}}}$ the average time of packet drop from the channel when the maximum number of retransmissions is exceeded.

The energy efficiency in packet transmission can be expressed in terms of the amount of energy consumed in transmitting each bit. Hence, the ratio of the amount of energy consumed unit per throughput represents the total energy consumption. In the study, the following graphics was generated for the two cases by calculating these analytical expressions (Fig. 2).

A comparison of the results shows that through this analytical model, it is possible to accurately estimate the system throughput and energy consumption. The IEEE 802.11ah system, which includes a restricted access window (RAW) mechanism, was also studied in the paper and compared with the basic analytical expression. The results obtained show that the RAW mechanism in a high-load network can significantly improve the performance of the system in terms of bandwidth, packet delay, and energy consumption characteristics. Based on the research results, it was concluded that the IEEE 802.11ah system has a high potential for use in M2M communication and IoT applications as a comprehensive, inexpensive, and energy-saving technology.



(a)

(b)

Fig. 2 **a** average throughput and **b** energy efficiency graph of the IEEE 802.11 network standard in different cases of restricted access window

3.3 WSN Built on Bluetooth

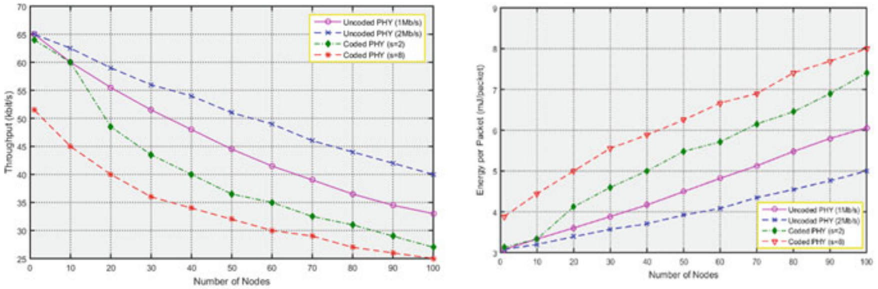
Bluetooth (IEEE 802.15.1 standard) is a wireless technology that operates in the ultra-high frequency range from 2402 to 2480 GHz and is used to transmit data over short distances between fixed devices and to create personal networks (PANs). It provides very low energy consumption, as well as a small size [25]. However, when many research works have been analyzed, this standard has hardly been used as a wireless sensor network for monitoring systems. Because monitoring systems are usually carried out in open areas, between objects at a relatively long distance from each other.

The performance effectiveness of this standard has been studied by many researchers. For example, in a research work by Behnam Badihi et al., "On the System-level Performance Evaluation of Bluetooth 5 in IoT: Open Office Case Study", the performance of the IEEE802.15.1 standard was studied on the basis of analytical and simulation models [26].

Due to the complexity of the new BLE5 protocol according to the IEEE 802.15.1 standard and the lack of system-level simulators, the researchers studied this protocol in detail. In the study, some important features of the BLE5 protocol were developed and results were obtained. The physical layer of the new IEEE 802.15.1 standard (BLE5) across the network has been studied based on end-to-end delay, power consumption, packet error rate, and bandwidth characteristics. However, changes in the network for different states of the physical layer have been studied, and the study has evaluated that the coded physical layer has low efficiency in high-load cases.

The following analytical expression is proposed for the network throughput, which represents the main characteristics of the network:

$$T_{\mathrm{BLE}}(n, s) = \frac{n * 8}{\frac{S_h}{R_b} + \frac{(n+s)R_c * 8}{R_b} + 2\tau + 2\max(t_{\mathrm{int}}, t_{tx} + t_{rx})} \tag{3}$$



(a) (b)

Fig. 3 **a** the average throughput and **b** the energy efficiency graphics of the end node of multi-node IEEE 802.15.1 network standard for different states of the physical layer

here:

n the amount of basic data in bytes transmitted from the coordinator to the managed node;
s the amount of basic data in bytes of the response packet;
R_b bitrate;
R_c code rate;
S_h header size in bits;
t_{tx} the time required to process the packet before it is transmitting;
t_{rx} the time required to process the package after it is received;
t_{int} inter-frame time;
τ latency of radio signal propagation.

Based on the above analytical expression, the following graphics were generated and the performance efficiency of the standard was evaluated.

Figure 3a shows that all physical channels except the coded physical channel with S = 8 can deliver 64 kbit/s of traffic to a pair of nodes. As the number of pair of nodes increases, a decrease in throughput can be observed in all cases of the physical channel. As expected, the throughput has high values for an uncoded physical channel with a speed of 2 Mbit/s. This situation can be explained by the fact that packets are sent in a short time, and therefore, they are less likely to collision with other neighbor nodes. The graphics show that when using the new IEEE 802.15.1 standard, it is advisable to select the mode that is most effective for the physical channel based on the network load and requirements.

3.4 WSN Based on ZigBee

ZigBee is a wireless networking technology based on the IEEE 802.15.4 specification, which includes a set of high-level communication protocols. This technology is used in energy monitoring, agricultural monitoring, traffic management systems, home automation, data collection on medical devices, and other similar low power and low-throughput personal communication networks [27, 28]. ZigBee technology is designed as a much simpler and cheaper technology than other wireless personal networks (WPANs), such as bluetooth or Wi-Fi. Depending on the output power and the environment, its low-power consumption limits the data transmission range to a radius of 10–100 m. ZigBee devices can transmit data over long distances, for which the network consists of several intermediate nodes. Intermediate nodes expand the service area of the network by retransmitting data. However, in this network, security issues are also taken into account, with strong protection (protected by 128-bit symmetric encryption keys) [29–31].

Numerous research works have been studied to evaluate the network efficiency of ZigBee technology. In one of them entitled “Performance evaluation of IEEE 802.15.4 with real-time queueing analysis” researched by Zhuoling Xiao et al., analyzed real-time buffering and virtual service time models that were not considered in other studies [29]. It is known that the buffering characteristic of a system that considered sleep mode different from other systems because in sleep mode packets from another node accumulate in the buffer, and the system initially has a high load when it switches from sleep mode to active mode [32]. The model proposed in this study consists of analyzing the network characteristics by dividing the superframe into an active period in the transmission slots and using an embedded discrete-time Markov chain in the system.

In this study, the following analytical expressions were identified and the results were obtained based on them:

1. **Throughput**:

$$T_{ZB} = \frac{S_p}{t_d} = \frac{l_{\text{pack}} \cdot \sum_{j,i,t_b} \pi_{(i,j)} p(t_b, j) p_s(t_b, j)}{t_B \cdot \left(\sum_{j,i,t_b} \pi_{(i,j)} p(t_b, j) t_v + \sum_j \pi_{(0,j)} \right)} \tag{4}$$

here:

S_p the average number of bits transmitted in each transition state;
t_d average time spent for each transition;
l_{pack} the number of bits corresponding to the packet;
t_b the duration of the backoff slot;
$\pi_{(i,j)}$ the steady-state probability of the Markov state defined in the buffering model;
$p(t_b, j)$ the probability of first channel sensing;
$p_s(t_b, j)$ the probability of successful access of the tagged node into the channel;
t_B the duration of the backoff slot;
t_v the virtual service time.

2. **Energy consumption**:

$$E_{ZB} = \frac{E_{\text{overal}}}{S_p} = \frac{E_{\text{succ}} + E_{\text{fail}} + E_{\text{idle}} + E_{\text{inap}}}{S_p} \tag{5}$$

where E_{succ}, E_{fail}, E_{idle}, and E_{inap} are the amounts of energy consumed in each state transition, respectively, represents the amount of energy consumed when the successful access of the node into the channel in the transition state (E_{succ}), the failure in starting the transmission in the transition state (E_{fail}), the state without connection to the channel (E_{idle}), and corresponds to the inactive period $\left(E_{\text{inap}}\right)$ and is determined by the following expressions.

$$E_{\text{succ}} = t_B \sum_{j=1}^{j_A} \sum_{i=1}^{L_m} \mathrm{E}\left[\pi_{(i,j)} p_s(t_b, j)\left(t_p P_{tx} + 2P_{rx} + t_b P_{\text{idle}}\right)\right] \tag{6}$$

$$E_{\text{fail}} = t_B \sum_{j=1}^{j_A} \sum_{i=1}^{L_m} \mathrm{E}[\pi_{(i,j)}(1 - p_s(t_b, j))\left(P_{tx} w(t) + P_{\text{idle}}\left(t_p + t_b + 2 - w(t)\right)\right)] \tag{7}$$

$$E_{\text{idle}} = t_B \sum_{j=1}^{j_A} \pi_{(0,j)} P_{\text{idle}} \tag{8}$$

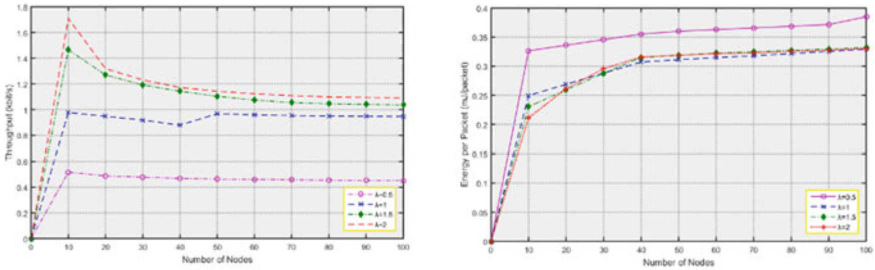
$$E_{\text{inap}} = t_B \sum_{j=t_A - l_{M-1}+1}^{j_A} \sum_{i=0}^{L_m} \mathrm{E}\left[\pi_{(i,j)} P_{slp}|_{t+t_b>t_A}\right] \tag{9}$$

here:

P_{tx}, P_{rx}, P_{idle}, and P_{slp}	the power consumption of the node during transmitting, receiving, idle and sleep mode, respectively;
j_A	number of slots in active period;
l_{M-1}	the length of (M-1)-th backoff stage;
L_m	buffer length in i-th slot;
t_p	transmission time.

Based on these expressions, values were determined and several graphics were built based on them. Below are the two main graphics used in the paper (Fig. 4).

When analyzing monitoring and control systems based on ZigBee technology, they proposed different architectures based on this technology. They are divided into three main parts for monitoring systems: sensor nodes that collect and transmit data, nodes that receive data, and a control center [33].



(a) (b)

Fig. 4 **a** average throughput and **b** energy efficiency graphics of the IEEE 802.15.4 network standard for different cases of packet arrival rate

4 Comparative Analysis of Wireless Technologies for WSN-Based Solar Power Station Monitoring

In the previous section, the applying of each wireless network technology used in wireless sensor networks, their advantages and disadvantages, as well as throughput and energy efficiency were analyzed on the basis of graphics. In this section, the above technologies are compared and the most optimal technology for the SPS monitoring system is selected.

WiMAX (IEEE 802.16) standard provides long distance, high bandwidth in data transmission like GSM standard [34, 35]. But today, the demand for it has almost disappeared due to the high cost of installation and maintenance, as well as the availability of standards such as its replacement GSM. Therefore, this technology is not used as WSN for monitoring systems.

The following table, compiled using the above analysis and the specifications of the wireless network technology, illustrates the capabilities of these technologies in more detail. The table shows the main technical characteristics of each wireless network technology used in WSNs (Table 1).

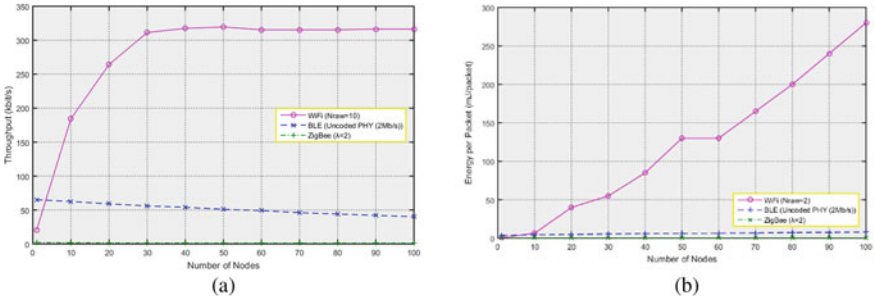
As can be seen from this table, three Wi-Fi, Bluetooth, and ZigBee technologies have the highest performance in the organization of local monitoring systems [36–38]. But, it is important to determine exactly which one of them has the most effective indicators for our research work. Figure 5 shows a comparative graphics of the throughput and energy efficiency of these network standards. This graphics is based on the values of the cases with the highest performance on the graphics given for each network standard in the previous section.

In terms of throughput, Wi-Fi technology showed the highest performance in the graphic. But in the second graphic, it can be seen that its energy consumption is high. Bluetooth technology shows average efficiency in terms of throughput. However, as the number of nodes increases, this characteristic of it decreases. This technology also has an average value in terms of energy consumption. As can be seen from the graphic, the lowest performance in terms of throughput belongs to ZigBee technology. But in terms of energy consumption, this technology has the lowest performance.

Table 1 Comparison of WSN technologies

Name/Characteristics	WiMAX	Wi-Fi (802.11)	Bluetooth (802.15.1)	ZigBee (802.15.4)
Data flow				
Data transfer speed	74 Mb/s	11, 54, 300 Mb/s	1–25 Mb/s	250 Kb/s
Max. throughput	10–100 Mb/s	7, 25, 100 Mb/s	2 Mb/s	150 Kb/s
Latency	100 ms	50 ms	100 ms	20–30 ms
Transmission distance	40 km	50–100 m	1–10 m	10–300 m
Characteristics of radio spectral performance				
Basic transmission frequency	2,3–13,6 GHz	2.4–5 GHz	2.4 GHz	868/915 MHz
Modulation type	OFDMA, TDD	MC-DSSS, CCK	FHSS	DSSS/ + TSCH
Channel bandwidth	5 MHz	22 MHz	1 MHz	2 MHz
Number of radio frequency channels	10–20	11–24	79	1,10,16
Tolerance to obstacles	High	High	Average	Low
Energy consumption				
Current consumption in sleep mode	–	50–70 mcrA	0.78 mcrA	4.18 mcrA
Max. current consumption	–	116 mA	30 mA	30–40 mA
Power consumption	1 Wt	835 mWt	215 mWt	36.9 mWt
Power efficiency	Low	Low	Average	High
Additional options				
Number of nodes by network size	1	32	2	Unlimited
Purpose of apply	Comprehensive voice, video and high size data transmission	Build a high-speed wireless local area network	Wired conductor replacement	Monitoring and control
Advantages	Existing infrastructure, high communication quality	High speed, flexibility	Easy to use, low cost	Reliable, low cost, low-power consumption

It is known that one of the main problems in remote monitoring systems is power supply. For this reason, monitoring systems typically use devices with the lowest energy consumption. In addition, sensors installed on monitored objects generate a very small data, and high-throughput technologies are not required for such systems. This means that the advantages of using ZigBee technology for monitoring systems can be evaluate high [39, 40].



(a)

(b)

Fig. 5 Comparison graphic of Wi-Fi, bluetooth and ZigBee wireless network standards in terms of **a** throughput and **b** energy efficiency

However, when using the ZigBee standard in WSNs, there are problems such as bypassing barriers, interference in the environment, frequency interference, sensitivity to noise, which are common to all other wireless network technologies [41, 42]. Despite the radio signal interference of other technologies in the same frequency range, such as Bluetooth and Wi-Fi, the ZigBee has 16-channel transmission, which ensures minimal interference compared to other standards. This technology also has the ability to manage network traffic.

5 Conclusion

Taking into account factors such as data transfer speed, a number of nodes, power consumption, the maximum distance between nodes, installation cost, installation complexity and overall reliability for monitoring systems of power supply sources of telecommunication facilities, ZigBee wireless technology has the best performance compared to technologies such as GSM, WiMAX, bluetooth, and Wi-Fi. The use of the ZigBee standard for WSN of monitoring systems compared to other radio modules, low-cost maintenance (compared to GSM and WiMAX), fastly connection between multiple endpoints (compared to bluetooth), and less power consumption (compared to Wi-Fi) and provides good configuration.

References

1. Davronbekov DA, Isroilov JD, Alimdjanov XF, Norkobilov SA, Axmedov BI (2021) Analysis of features of wireless sensor networks. Scientific collection «InterConf», (41): Proceedings of the 7th international scientific and practical conference «Scientific horizon in the context of social crises», 6–8 Feb 2021, Tokyo, Japan, Otsuki Press, pp 1044–1051
2. Khujamatov H, Toshtemirov T (2020) Wireless sensor networks based agriculture 4.0: challenges and apportions. 2020 international conference on information science and communications technologies (ICISCT). Tashkent, Uzbekistan. https://doi.org/10.1109/ICISCT50599.2020.9351411

3. Shu Y, et al (2014) Internet of things: wireless sensor networks. White paper, Geneva, Switzerland
4. Khujamatov KE, Khasanov DT, Reypnazarov EN (2019) Modeling and research of automatic sun tracking system on the bases of IoT and Arduino UNO. International conference on information science and communications technologies ICISCT 2019. Tashkent, Uzbekistan. https://doi.org/10.1109/ICISCT47635.2019.9011913
5. Khujamatov K, Khasanov D, Reypnazarov E, Akhmedov N (07–09 Oct 2020) Networking and computing in internet of things and cyber-physical systems. The 14th IEEE international conference application of information and communication technologies. Tashkent, Uzbekistan. https://doi.org/10.1109/AICT50176.2020.9368793
6. Davronbekov DA, Aliev UT, Isroilov JD, Alimdjanov XF (2019) Power providing methods for wireless sensors. International conference on information science and communications technologies ICISCT 2019. Tashkent, Uzbekistan
7. Khujamatov K, Ahmad K, Reypnazarov E, Khasanov D (2020) Markov chain based modeling bandwith states of the wireless sensor networks of monitoring system. Int J Adv Sci Technol 29(4):4889–4903. http://sersc.org/journals/index.php/IJAST/article/view/24920
8. Siddikov IK, Khujamatov KE, Khasanov DT, Reypnazarov ER. Modeling of monitoring systems of solar power stations for telecommunication facilities based on wireless nets. Chemical technology, control and management. Int Sci Techn J 2020(3):4. https://uzjournals.edu.uz/ijctcm/vol2020/iss3/4
9. Gao D-Y, Zhang L-J, Wang H-C (2011) Energy saving with node sleep and power control mechanisms for wireless sensor networks. J China Univ Posts Telecommun 18(1):49–59
10. Khujamatov KE, Khasanov DT, Reypnazarov EN (2019) Research and modelling adaptive management of hybrid power supply systems for object telecommunications based on IoT. International conference on information science and communications technologies ICISCT 2019. Tashkent, Uzbekistan. https://doi.org/10.1109/ICISCT47635.2019.9011831
11. Siddikov IX, Sattarov KA, Khujamatov KE (2–4 Nov 2017) Modeling of the transformation elements of power sources control. International conference on ICISCT—2017. Tashkent, Uzbekistan
12. Siddikov IK, Sattarov KA, Khujamatov KE (2016) Research of the influence of nonlinear primary magnetization curves of magnetic circuits of electromagnetic transducers of the three-phases current. Universal J Electr Electron Eng 4(1):29–32. Horizon Research Publishing Corporation, USA. https://doi.org/10.13189/ujeee.2016.040104
13. Davronbekov DA, Muxamedaminov AO, Axmedov BI (2020) The role of wireless networking technology today. Инновационные научные исследования: Теория, Методология, Практика". Сборник статей XX Международной научно-практической конференции. с. 77–79
14. Davronbekov DA, Rakhimov BN, Alimdjanov XF, Axmedov BI (2021) Review of wearable wireless sensor network. Scientific collection «InterConf», (41): proceedings of the 7th international scientific and practical conference «Scientific horizon in the context of social crises», 6–8 Feb 2021, Tokyo, Japan, Otsuki Press, pp 1052–1058
15. Bhoyar P, Sahare P, Dhok SB, Deshmukh RB (2018) Communication technologies and security challenges for internet of things: a comprehensive review. Int J Electron Commun 99(2019):81–99
16. Lubobya SC, Dlodlo ME, De Jager G, Zulu A (2015) Throughput characteristics of WiMAX video surveillance systems. International conference on advanced computing technologies and applications (ICACTA-2015), Procedia Comput Sci 45:571–580
17. Davronbekov DA, Matyokubov UK (2020) The role of network components in improving the reliability and survivability of mobile communication networks. Acta Turin Polytech Univ Tashkent 10(3):Article 2, 7–14
18. Matyokubov UK, Davronbekov DA. Approaches to the organization of disaster-resistant mobile network architecture in Uzbekistan. Acta Turin Polytech Univ Tashkent. Выпуск 2/2020, С. 34–42

19. Siddikov IK, Sattarov KA, Khujamatov KE (4–5, 2018) Modeling and research circuits of intelligent sensors and measurement systems with distributed parameters and values. Chem Technol Control Manag, Int Sci Tech J, Tashkent 50–55
20. Ferdousi A, Enam F, Khan SR (Aug 2013) The performance evaluation of IEEE 802.16 physical layer in the basis of bit error rate considering reference channel models. Int J Cybern Inform (IJCI) 2(4):17–26
21. Matyokubov UK, Davronbekov DA. Increasing energy efficiency of base stations in mobile communication systems. Acta Turin Polytech Univ Tashkent. Выпуск 1/2020, C.22–27
22. Lin H-H, Shih M-J, Wei H-Y, Vannithamby R (2015) Deep sleep: IEEE 802.11 enhancement for energy-harvesting machine-to-machine communications. Wirel Netw 21(2):357–70.
23. Shaaban S, El Badawy HM, Hashad A (2–4 July 2008) Performance evaluation of the IEEE 802.11 wireless LAN standards. Proceedings of the world congress on engineering 2008, vol I. WCE 2008, London, UK
24. Raeesi O, Pirskanen J, Hazmi A, Levanen T, Valkama M. Performance evaluation of IEEE 802.11ah and its restricted access window mechanism. ICC'14–W7: workshop on M2M communications for next generation IoT
25. Tosi J, Taffoni F, Santacatterina M, Sannino R, Formica D (2017) Performance evaluation of bluetooth low energy: a systematic review. Sensors 17:2898, 1–34. https://doi.org/10.3390/s17122898
26. Badihi B, Ghavimi F, Jantti R. On the system-level performance evaluation of bluetooth 5 in IoT: open office case study. ©2019 IEEE
27. Khujamatov K, Khasanov D, Reypnazarov E, Akhmedov N (2021) Existing technologies and solutions in 5G-enabled IoT for industrial automation. Springer Nature Switzerland AG 2021. https://doi.org/10.1007/978-3-030-67490-8_8
28. Khujamatov H, Khasanov D, Reypnazarov E, Akhmedov N (2020) Industry digitalization concepts with 5G-based IoT. 2020 international conference on information science and communications technologies (ICISCT). Tashkent, Uzbekistan. https://doi.org/10.1109/ICISCT50599.2020.9351468
29. Khujamatov H, Reypnazarov E, Lazarev A (2021) Modern methods of testing and information security problems in IoT. Bulletin of TUIT: management and communication technologies 4:Article 4
30. Di Marco P, Skillermark P, Larmo A, Arvidson P, Chirikov R (June 2017) Performance evaluation of the data transfer modes in bluetooth 5. IoT and machine-type communication, IEEE communications standards magazine, pp 92–97. https://doi.org/10.1109/MCOMSTD.2017.1700030
31. Cuomo F, Abbagnale A, Cipollone E (2013) Cross-layer network formation for energy-efficient IEEE 802.15. 4. ZigBee wireless sensor networks. Ad Hoc Netw 11(2):672–86
32. Xiao Z, Zhou J, Yan J, He C, Jiang L, Trigoni N (2018) Performance evaluation of IEEE 802.15.4 with real time queueing analysis. Ad Hoc Networks 73:80–94. https://doi.org/10.1016/j.adhoc.2018.01.006
33. Lee B-H, Yundra E, Wu H-K, Udin Harun Al Rasyid M (2015) Analysis of superframe duration adjustment scheme for IEEE 802.15.4 networks. EURASIP J Wireless Commun Networking (2015) 103:1–17. https://doi.org/10.1186/s13638-015-0296-3
34. Khujamatov H, Reypnazarov E, Hasanov D, Nurullaev E, Sobirov S (Dec 2020) Evaluation of characteristics of wireless sensor networks with analytical modeling. Bulletin of TUIT: management and communication technologies bulletin of TUIT: management and communication technologies, vol 3. https://uzjournals.edu.uz/tuitmct/vol4/iss1/4
35. Muradova AA, Khujamatov KE (2019) Results of calculations of parameters of reliability of restored devices of the multiservice communication network. International conference on information science and communications technologies ICISCT 2019. Tashkent, Uzbekistan
36. Khujamatov H, Toshtemirov T, Khasanov D, Saburova N, Xamroyev I (2021) IoT based agriculture 4.0: challenges and opportunities. Bull TUIT: Manag Commun Technol 4:5
37. Gomez C, Oller J, Paradells J (2012) Overview and evaluation of bluetooth low energy: an emerging low-power wireless technology. Sensors 12(9):11734–11753

38. Di Marco P, Skillermark P, Larmo A, Arvidson P, Chirikov R (2017) Performance evaluation of the data transfer modes in bluetooth 5. IEEE Commun Stand Mag 1(2):92–97
39. Ray PP, Agarwal S (2016) Bluetooth 5 and internet of things: potential and architecture. In: 2016 international conference on signal processing, communication, power and embedded system (SCOPES). IEEE, pp 1461–1465
40. Khujamatov H, Reypnazarov E, Akhmedov N, Khasanov D (2020) IoT based centralized double stage education. 2020 international conference on information science and communications technologies (ICISCT). Tashkent, Uzbekistan. https://doi.org/10.1109/ICISCT50599.2020.9351410
41. Fourty N, Van Den Bossche A, Val T (2012) An advanced study of energy consumption in an IEEE 802.15.4 based network: everything but the truth on 802.15. 4 node lifetime. Comput Commun 35(14):1759–67
42. Khujamatov H, Reypnazarov E, Akhmedov N, Khasanov D (2020) Blockchain for 5G Healthcare architecture. 2020 international conference on information science and communications technologies (ICISCT). Tashkent, Uzbekistan. https://doi.org/10.1109/ICISCT50599.2020.9351398

An Energy Efficient Routing for Emergency Rescue in IoT-Based WSN

J. Shreyas, S. Shilpa, P. K. Udayaprasad, N. N. Srinidhi, and S. M. Dilip Kumar

Abstract The rapid expansion demand of sensor devices have attained significant attention for wireless sensor network (WSN) and demands many Internet of Things (IoT) applications for real-time usage. The routing technique must capable of efficiently delivering packets to sink to achieve large-scale IoT application's reliability. How to guarantee real-time emergency reaction capabilities during data transmission is now a difficult topic for research under the demand of large IoT network scale expansion. Existing routing scheme fails to deliver data and has higher routing latency due to transmissions failures and lower energy nodes. In this article, an energy efficient emergency response routing scheme (EEER) is proposed. The proposed EEER delivers the critical data efficiently under the emergency with lower latency. Extended transmission range covers wide range of intermediate nodes for data communication. This scheme selects nodes with higher residual energy for forwarding emergency data towards sink. To ensure this, the output of the proposed device is validated with an comprehensive simulation test. Results tested using NS2 simulation method indicate an improvement of 5% decrease in energy usage, 6% increase in packet delivery ratio and 7% decrease in end to end delay.

Keywords Energy aware · Latency · Emergency data · Internet of Things

1 Introduction

To employ wide sensor networks in the Internet of Things (IoT) is to link every object with a distinct identity up in the physical world [1, 2]. The primary idea behind the IoT is to allow a variety of devices among us to connect with one another, such as cell phones, actuators, gadgets, smart homes and certain more devices [3]. IoT-related innovation brings people closer to the real world and gives them with Context-Aware information relying on real data from each sensor node [4, 5].

J. Shreyas · S. Shilpa · P. K. Udayaprasad (✉) · N. N. Srinidhi · S. M. Dilip Kumar
Department of Computer Science and Engineering, UVCE, Bengaluru, India
e-mail: udayaprasad43@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_23

In IoT applications, the sensed data has to be routed to different destination for further operations, which is achieved by designing an efficient routing protocol [6]. During communication, node cooperate with each other and forwards data to destination. Re-transmission is a viable idea for increasing dependability, but it can also increase communication delays and impact real efficiency [7, 8]. The routing protocol's ability to offer real efficiency is the challenging issue. The routing protocol must be very reliable, and the responsibility of each node should be taken into consideration totally [9–11].

Existing system mainly concentrates on minimizing delay on a single path and does not considers energy levels of nodes. More retransmission causes congestion and leads to packet loss, hence this inspired us to present an energy efficient emergency response routing scheme (EEER) which prevents unnecessary retransmission delay. This scheme considers average path delay and energy levels during data transmission and guarantees higher packet deliver to sink through reliable routing.

The problem considered in this research article is to propose a routing mechanism which promises the improvement in the lifetime of the IoT network based on WSN by considering the below objectives.

1. To enhance the lifetime of the network.
2. To better utilize the energy consumption.
3. To reduce the transmission delay.

The contributions of the paper are as follows:

1. The routing algorithm based on energy efficiency is proposed.
2. Dijkstra and Transmission adjustment procedure is used to find the optimal alternative route.
3. The proposed work is evaluated using NS2 simulation toll, and the obtained results are compared against ERGID method.

2 Related Works

In [12], author presented numerous analyses on real-time routing systems which have been presented in recent years. And the majority of issues are focused data transfer issues. First, the framework must maintain real-time packet accuracy while reducing the amount of missing packets due to mislaying and retardment. Second, the mechanism must regulate network energy utilization and prevent some nodes from dying prematurely.

In [13], author presented an instantaneous data transmission that takes advantage of both the broadcasting characteristic and the notion of spatial opportunity. In general, nodes having a shorter length allowed queue have a stronger sending priority and are quite likely to be chosen as the transmitting node.

In [14], proposed method to determining transmission rates of neighbour nodes is required to determine the timeliness of data streams from the originating node to the recipient node. Route scan be included to the potential set of nodes if the data transference cost is more than the overall cost. The protocol utilizes spatial forwarding technique, wherein the node obtains location features for routing configuration from its neighbours, providing better network routing and significant efficiency.

In author proposed a routing scheme that modifies the SPEED methodology to provide numerous and multi-way communication. SPEED seeks to increase data transmission dependability by providing two services such as instantaneous efficiency and durability. Depending on the channel conditions, packets could select the optimal solution. The use of a multi-path transmission approach improves the chances of eliminating a blank region [15].

3 Proposed System

3.1 Network Module

In the network module, uniform deployment of nodes with similar communication range in a sensing region of the network Sink node (destination node) collects the emergency data from lower levels and process the information for emergency response. Each node calculates its one hop neighbour using euclidean distance (Fig. 1).

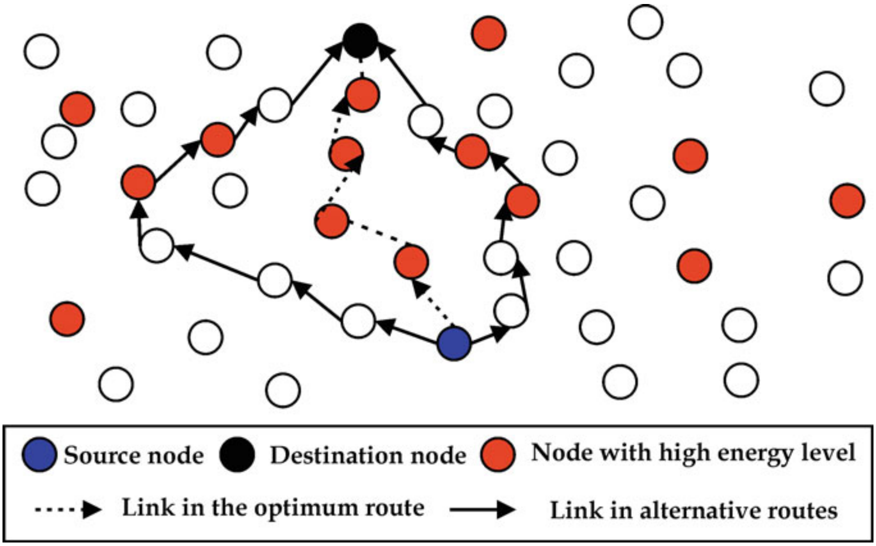

Source node
Destination node
Node with high energy level
Link in the optimum route
Link in alternative routes

Fig. 1 Network topology

3.2 Energy Module

The energy dissipation of nodes, in delivering data, and receiving data is very important in network. Energy consumed by node for wireless transmission and reception per bit of a circuit is given as e_{elec}. e_{amp} is the amplified amount of energy, the utilization of energy function is given as :

$$e_{t_x}(k, d) = e_{\text{elec}}(k) + e_{\text{amp}}(k, d) = e_{\text{elec}} \times k + e_{\text{amp}} \times k \times d^2 \tag{1}$$

To energy required to receive data at the receiver is given as

$$e_{r_x}(k) = e_{t_x}(k) = e_{\text{elec}} \times k \tag{2}$$

Residual energy is presented as

$$e_{\text{res}} = \text{IE} - (e_{t_x}(k, d) + e_{r_x}(k)) \tag{3}$$

where IE is initial energy allocated initially to nodes.

4 Proposed EEER Method

To overcome the issue of valid routes being ignored, we implement a technique called Delay Iterative Method, which would be reliant on delay estimates. The DIM technique places the node in the potential set based on the delay value. At same time, neighbour connectivity updates the routing information table on a regular basis to guarantee real-time performance. From the neighbour list, the method identifies the nodes with the largest sensing regions as coverage network nodes. If multiple nodes have the same sensing range, choose one at random to be the coverage active neighbour by dynamic adjustment of transmission range to cover maximum intermediate node for reliable and energy efficient routing of emergency data.

5 Performance Analysis

The performance of the EEER technique is simulated on NS2 simulator, and the obtained results are contrasted against traditional emergency response protocol and parameters considered for evaluation are energy consumption, overhead and packet delivery ratio. Table 1 gives the parameters and their values used in the proposed work experimentation.

Table 1 Parameters and their values of experimentation

Parameter	Value
Number of nodes	50–100
Node's initial energy	100 J
Network queue size	50
Simulation run-time	50 s for each round
MAC communication	802.11
Node formation	Grid deployment
Network plot area	800 × 800 m
Network traffic type	CBR
Size of data packet	512 bytes

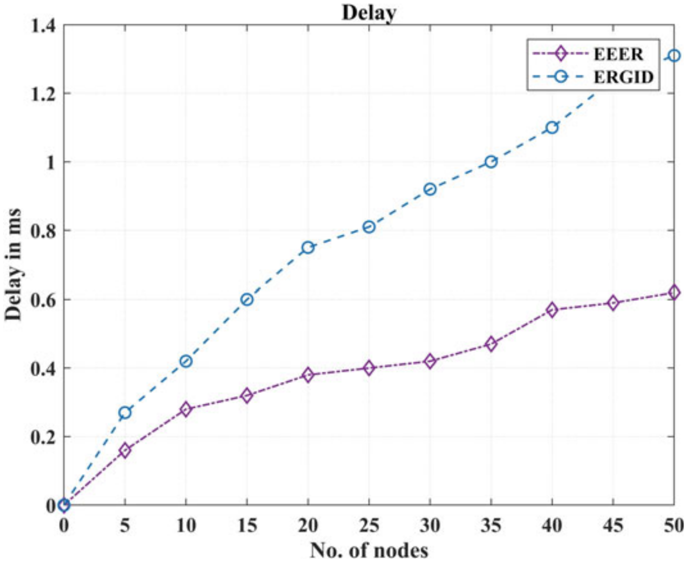


Fig. 2 Delay versus No. of nodes

5.1 End to End Delay

As shown in Fig. 2, the e2e delay graph can be observed from graph that the proposed EEER has decreased delay compared to existing ERGID routing scheme, this is due to the paths selection in EEER is based on nodes having higher remaining energy and the higher coverage of neighbour nodes. This enables the routing scheme to select paths with lesser delay and energy efficient to forward emergency packets to sink.

Algorithm 1 EERO Algorithm

```
procedure
    Input: Source (S), Destination D)
    Output: S $\rightarrow$ D (Shortest Path)
    Network constructed in Graph format
    Apply Dijkstra algorithm for routh discovery

    Dijkstra Procedure
    G-Graph, V-Vector, P-Path, Dijkstra(G, S)

    for (each_V in G) do
        dist_[V] = $\infty$
        prev_[V] = NULL
        if (Current V $\neq$ S) then
            add V neighbours to pri_queue
            dist_[S] = 0
            U = Calculate MIN neighbour from pri_queue
                end if
        end for

    Transmisson adjustment Procedure
    for (nodes in the pri_queue) do
        select nodes with larger sensing area
        compute energy_consumption $e_{t_x}(k, d)$
        select nodes with higher residual energy $e_{res}$
        end for
    Forward emergency data to sink
end procedure
```

5.2 *Packet Delivery Ratio (PDR)*

Figure 3 shows the proposed EEER technique PDR compared with existing ERGID method. It is seen from graph that the proposed EEER scheme has higher percentage of packets delivered at sink compared to ERGID. EEER maintains the path stability until the packets are delivered at sink. However, ERGID has lower path stability and has lower packet delivery ratio.

5.3 *Energy Consumption*

In Fig. 4 shows the utilization of energy of proposed EEER technique with conventional ERGID. It is observed that the proposed EEER scheme consumes less energy than ERGID. EEER considers the average energy of nodes in the network and tries to balance network energy and chooses the path with higher residual energy for energy efficient paths and improves network lifetime.

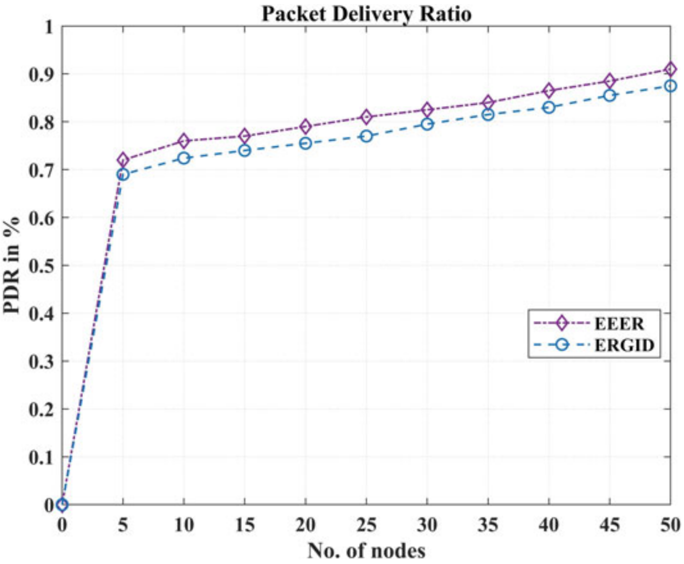


Fig. 3 PDR versus no. of nodes

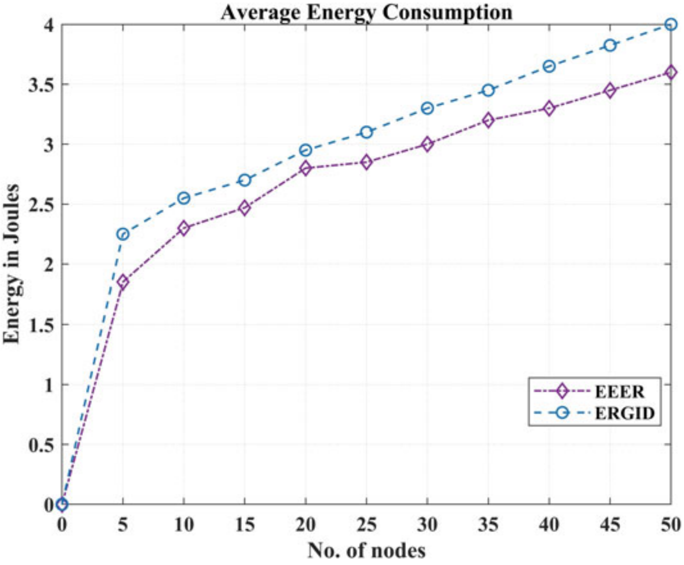


Fig. 4 Energy versus no. of nodes

6 Conclusion

Routing emergency data to sink is considered as much priority for rescue operations, data has to reach to sink with lower latency to avoid any incidents. In this article, an energy EEER technique is presented to route emergency data for rescue operations. This scheme considers energy efficient routes based on the nodes residual energy and selects optimal neighbours to carry emergency data toward sink. This scheme also dynamically adjusts nodes transmission range and balances the nodes energy to increases the lifetime of the network. Proposed scheme is compared with existing scheme and network parameters like delay, PDR and energy consumption is evaluated.

References

1. Shreyas J, Jumnal A, Kumar SD, Venugopal KR (2020) Application of computational intelligence techniques for internet of things: an extensive survey. Int J Comput Intell Stud 9(3):234–288
2. Stankovic JA (2014) Research directions for the internet of things. IEEE Internet Things J 1(1):3–9
3. Naeem F, Tariq M, Poor HV (2020) SDN-enabled energy-efficient routing optimization framework for industrial internet of things. IEEE Trans Industr Inform 17(8):5660–5667
4. Shreyas J, Singh H, Tiwari S, Srinidhi NN, Kumar SD (2021) CAFOR: congestion avoidance using fuzzy logic to find an optimal routing path in 6LoWPAN networks. J Reliable Intell Environ pp 1–16
5. Haque KF, Abdelgawad A, Yanambaka VP, Yelamarthi K (2020, June) An energy-efficient and reliable RPL for IoT. 2020 IEEE 6th world forum on internet of things (WF-IoT). IEEE, New York, pp 1–2
6. Hameed AR, ul Islam S, Raza M, Khattak HA (2020) Towards energy and performance-aware geographic routing for IoT-enabled sensor networks. Comput Electr Eng 85:106643
7. Yarinezhad R, Azizi S (2021) An energy-efficient routing protocol for the internet of things networks based on geographical location and link quality. Comput Networks 193:108116
8. Njah Y, Cheriet M (2021) Parallel route optimization and service assurance in energy-efficient software-defined industrial IoT networks. IEEE Access 9:24682–24696
9. Khan IU, Qureshi IM, Aziz MA, Cheema TA, Shah SBH (2020) Smart IoT control-based nature inspired energy efficient routing protocol for flying ad hoc network (FANET). IEEE Access 8:56371–56378
10. Sankar S, Ramasubbareddy S, Chen F, Gandomi AH (2020, December) Energy-efficient cluster-based routing protocol in internet of things using swarm intelligence. In: 2020 IEEE symposium series on computational intelligence (SSCI). IEEE, New York, pp 219–224
11. Kaur G, Chanak P, Bhattacharya M (2021) Energy efficient intelligent routing scheme for IoT-enabled WSNs. IEEE Internet Things J
12. Oh S, Yim Y, Lee J, Park H, Kim S-H (2013) An opportunistic routing for real-time data in Wireless sensor networks. In: Wireless communications and networking conference (WCNC). Shanghai, China, pp 1157–1162
13. He T, Stankovic J, Lu C, Abdelzaher T (2003) SPEED: a stateless protocol for realtime communication in sensor networks. In: Proceedings of the 23th IEEE international conference on distributed computing systems, Providence, RI, US, pp 46–55
14. Lee EFC, Ekici E (2006) MMSPEED: multipath multi-SPEED protocol for QoS guarantee of reliability and timeliness in wireless sensor networks. IEEE Trans Mob Comput 5(6):738–754
15. Shreyas J, Kumar SM (2019, November) A survey on computational intelligence techniques for internet of things. In: International conference on communication and intelligent systems. Springer, Singapore, pp 271–282

An Energy-Efficient Data Sensing Technique Using Compressive Sensing for IoT-Based Systems

Amarjeet Kaur and Prakash Kumar

Abstract Internet of Things (IoT) is an emerging technology that interconnects real-world devices using cyber world systems. There are various sensing nodes that collect data from real-world environments. These nodes deployed at various geographical locations and send sensed data for further processing. Sensor nodes consume energy to transmit data. Usually, the sensing devices are battery-operated; hence, the more they sense and transmit data, the more the energy is depleted. Therefore, efficient energy consumption is an important issue, especially in IoT systems. Energy is mainly consumed in bit transmission. If the data size is reduced, energy consumption is also reduced and the network lifespan also gets prolonged. In this paper, we have used a compressive sensing technique to reduce data size at sensor nodes; consequently, there is a significant reduction in energy consumption. Experimental results show that a significant amount of energy is saved when compressive sensing is used while sensing and transmitting data.

Keywords Internet of things (IoT) · Energy-efficient · Sensing Layer · Compressive Sensing

1 Introduction

The Internet of Things (IoT) is an emerging area that connects number of physical objects (things) anytime, anywhere, with anything and anyone using Internet/network and bridges the gap between physical and cyber world. In IoT, objects/things sense the environment, collect data, and communicate with each other and provide useful information that is helpful in understanding the physical/real-world and identifying irregularities immediately [1]. The physical objects/things include sensors, actuators, hardware, software, storage, etc, that are used in various fields such as transport, industry, healthcare, and home appliances [2]. Apart from these, there are various applications of IoT in different areas like smart agriculture, smart home,

A. Kaur (✉) · P. Kumar
Department of CSE, Jaypee Institute of Information and Technology, Noida, India
e-mail: amarjeet.kaur@jiit.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_24

monitoring environmental parameters, weather forecasting, healthcare, smart waste management, smart city, etc. [3]

The IoT applications are facing lots of challenges [4] like hardware miniaturization, bandwidth utilization, heterogeneity device, energy constraint, energy harvesting, privacy, and security, etc. Energy constraint is a major issue for IoT based systems [5, 6]. IoT devices are power backed by limited battery capacity that is used for real-time data collection and transmission [6]. The more data collection and transmission, the more battery power/energy consumption is there [6]. Hence, lifetime [7] of IoT-based system depends on the residual energy of individual devices. Reducing energy consumption of device prolong the lifetime for proper functioning of IoT-based systems is still a great challenge. It is still an unattended area that highly motivates us to work on energy-efficient perception layer for IoT-based systems.

In this paper, our objective has been to reduce energy consumption of sensor nodes (SNs) in communication at sensing layer as we knew that SNs consume more energy in communication as compared to sensing and processing. In communication, energy consumption is due to transmission of each data/message bit. If data/message size reduces in form of bits, energy consumption will be reduced for messages due to a smaller number of bits. In wireless sensor networks and IoT systems, compressive sensing is used to reduce number of bits in data/message. Therefore, we proposed an energy-efficient sensing layer where SN acquire data in compressed form and reduce data size which save transmission energy and prolong the lifespan of SNs.

Further, the paper is organized as follows. Sect. 2 investigates the background and related work pertaining to architecture of IoT based systems using compressive sensing techniques. An energy-efficient sensing layer is proposed in Sect. 3. Sect. 4 presents its simulation and performance analysis and section. Finally, Sect. 5 concludes the paper.

2 Background and Related Work

Designing an energy-efficient IoT-based system is a challenging issue because of battery constraints of the devices at sensing layer. Therefore, many authors addressed these issues and proposed various solutions (deployment schemes, sleep mode, etc.) for this.

In [8], uses a "self-organized things" concept to save energy where sensor nodes configure, optimize and use healing mechanism automatically. Sensor nodes were put into sleep mode to save energy whereas coverage area were compromised.

An "energy-efficient index tree" (EGF-tree) is proposed to save energy usage in collecting, querying, and aggregating data from sensor nodes in different area [9]. Sensor nodes were arranged in a tree structure. Queries were sent by the query manager from sensor nodes to the base station and vice versa in an energy-efficient way. Similar to [9], a "clustering index tree" (ECH-tree) is proposed [10], in this, sensor nodes were divided into grid cells. These grid cells form an energy-efficient hierarchical ECH tree. The sensed data were transmitted only when there was a significant difference in current and previous value by using time-correlated region technique which saves energy. An "object group mobility (OGM)" method was proposed where many objects move together in a vehicle or are carried by a human being where these objects can be grouped [11]. It improves the location accuracy and reduces traffic signals. The group formation and spatial correlation is used to reduce energy consumption. In [12], sleep mode is used to save energy of the user equipment. It proposed how to maximize the sleep period without compromising the quality-of-service (QoS) to save energy of nodes. A hierarchical three-layer arrangement of sensor nodes was proposed for Industrial IoT in [13] to optimize energy consumption and balance the traffic. Sleep mode was used for sensor nodes to save energy and prolong the network life. In [7], the same type of architecture is proposed for general IoT systems. Here too, sleep mode is used to save energy consumption and sleep interval is predicted by considering the number of parameters like previous usage history, their remaining battery level, and quality of information required for a particular application etc.

Compressive sensing (CS) is an energy-efficient data sensing technique that delivers the complete data with fewer data samples as compared to the original data samples when sensed data is sparse or compressible in some domain. In WSN, where energy is again a constraint for sensor nodes, compressive sensing is used to sense and collect data in different proposed techniques and applications. In [14–17], an energy-efficient data collection and aggregation is done with compressive sensing. In [18–20], an optimal routing tree is constructed to route data from SNs to base station with objective of energy-efficient data collection in network and optimal route is constructed with sparse random measurement. In [14–20], compressive sensing is used for data sensing, collection, and aggregation to save energy. In visual sensor network (VSN), compressive sensing is used to reduce data size for energy saving and efficient bandwidth usages in [21]. Image compressive sensing [22] is used to decrease the size of images for green Internet of Things because it used low power and bandwidth. In [23], author used compressive sensing to design multi-class privacy-preserving cloud computing scheme (MPCC), this decreases the issue of energy, and data transmission as well as storage caused by massive IoT sensor data. To reduce energy consumption, compressive sensing is used to collect sensory data in IoT networks [24]. A multicluster cooperative CS (CCS) scheme is proposed in [25], which is used for large-scale IoT networks for observation of the environment.

All of the aforementioned studies [7, 11–13] discussed architectures and deployment schemes. All of them use a group or hierarchical structure of nodes for sensing layer to save energy. In [14–26], compressive sensing is used for efficient utilization of bandwidth and reduce energy consumption by reducing data size. In this paper, combined both and proposed a structure organization of nodes and compressive sensing for efficient energy utilization.

3 Proposed Model

3.1 Compressing Sensing (CS)

In CS, very few data samples are collected. Further original data is reconstructed from these samples at receiver end. It is a two-stage paradigm where data sensing and compression is done in a single step. The benefit of CS is that it reduces the data size because of limited data samples which further reduces communication delay, energy drain and efficiently uses the bandwidth. At sensing layer of IoT, the energy is mainly consumed in wireless communication rather than processing. As per state-of-art literature, radio transmitters consume energy in nJ/bit and other circuit of sensor node consume energy in pJ/bit [26]. The mathematical equation for CS is:

$$\mathbf{y} = \phi\mathbf{x} \tag{1}$$

where **x** is a sparse signal, ϕ is sensing or measurement matrix to compress data, and **y** is compressed data. **x** should be sparse signal or we have to make it sparse before compression. The original data is reconstructed from compressed data at receiver.

As per literature [7–20], IoT based systems are divided into three function/operational layers, viz, sensing, processing, and application as shown in Fig. 1. The sensing layer senses data from environment and sends it to processing layer where this raw data is processed and created a useful information for application layer.

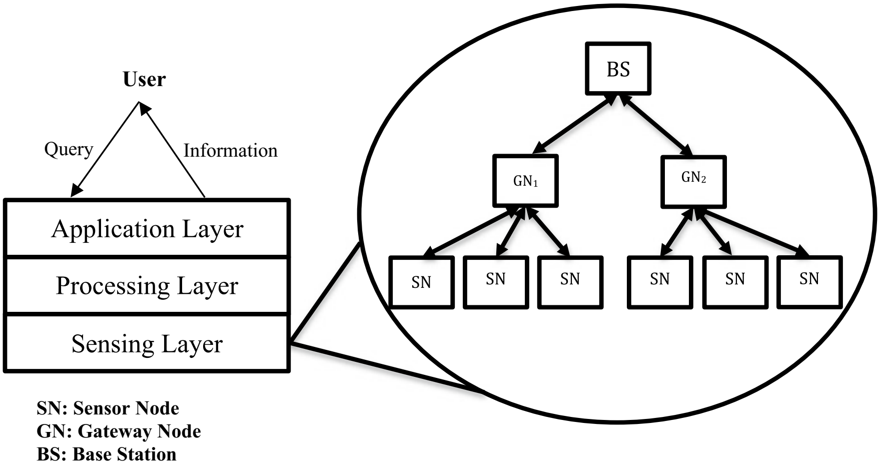


Fig. 1 IoT based system's layers and sensing layer

3.2 Network Model

Three type of nodes are there at sensing layer: sensor nodes (SNs), gateway nodes (GNs), and base station (BS) as shown in Fig. 1. SNs sense data at a regular interval from environment and send this data to GNs. It is assumed that GNs are one hop away from SNs. GNs receive data from SNs and forward it to BS without any processing. GNs work as relay nodes. Initially, SNs are assigned to GN on the basis of Euclidean equation which is used to calculate distance between any SN and GN.

3.3 Energy Model

The devices are energy-constrained at the sensing layer, especially SNs which are continuous or at regular interval sense data. The energy is consumed in three tasks, namely, data sensing, processing, and communication (transmitting and receiving). Energy consumption in sensing and processing is very less as compare to energy consumed in communication [27]. Therefore, only communication energy consumption is considered and calculated as per Friis free space model [15]. As per Friis free space model, following two equations represent the energy consumption in communication:

$$E_{tx} = \left(E_{\text{elec}} + \varepsilon_0 * d^2\right) * L \tag{2}$$

$$E_{rx} = \left(E_{\text{elec}} * L\right) \tag{3}$$

where E_{tx} and E_{rx} energy consumption at a SN for data transmission and receiving respectively. E_{elec} is energy consumed by radio electronics. ε_0 is transmit amplifier of SN. d is distance between SN and GN, and L is length of data send from SN to GN.

In this proposed model, only energy consumption for SNs is considered because we are studying compressive sensing and energy consumption here. Therefore, it is assumed that GNs and BS have adequate energy for working. To save energy compressive sensing is implemented with this energy model. The compressive sensing reduces number of bits of data to be transmitted (as shown in equation (1)) which reduces energy consumption in communication as per equation (2) and (3).

4 Simulation and Performance Analysis

In this section, reduction of energy consumption is validated using compressive sensing that has been discussed in proposed model. Further, Simulation is done in

Table 1 Parameters and their values

Parameter	Values
d	10 m < d < 100 m
L	10–250 kbps
E_{elec}	50 nJ/bit
ε_0	100 pJ/bit/m^2
Time interval	5 s
Protocol	*Zigbee*
Bandwidth	250 kbps
Range of data size	10–250 kbps
Time of simulation	24 h

python. Sensor nodes are randomly deployed in 100×100 m^2 area, in which there is one GN and one BS. SNs are varied from 5 to 455 in this area. The parameters used to configure this network are shown in Table1. Zigbee protocol is used for this experiment; therefore, the distance between SNs and GN is considered 10–100 m as the range of zigbee protocol is 100 m. It is assumed that all SNs are homogenous and have same energy values are set, same data generation rate (10–250 kbps) is set. $E_{\text{elec}} = 50$ nJ/bit *and* $\varepsilon_0 = 100$ pJ/bit/m^2 for sensing nodes. Experiment time considered is for 24 h and each SN sends data to GN after every 5 Secs. Further, variation of energy consumption is examined with compressive sensing and without compressive sensing by varying the number of sensing nodes. Usually, compressive sensing reduces data size up to 50% by using compressive algorithm at SNs [27, 28]. In our work energy, consumption of SNs only is considered and reconstruction is not considered here as reconstruction of signal will be done at BS.

Figures 2 and 3 show graph between number of SNs and data sent to GN by these SNs. SNs send data after every 5 sec intervals. Graph shows the data sent by SNs, where all SNs, a few random SNs (40% of SNs and 70 % of SNs), and none of SNs use compressive sensing. From Figs. 2 and 3 it is seen that as the number of SNs are increased the sensed data sent by them is also increased which is natural. At the same time, it is also seen that if compressive sensing is implemented on all SNs or on few random SNs for data sensing, the transmitted data reduces accordingly. Therefore, if compressive sensing is used on SNs, data for transmission can be reduced.

Graphs in Figs. 4 and 5 is drawn between number of SNs and energy consumed by them to transmit data to GN. Similar to the previous one, this graph also shows energy consumed by all SNs, a few random SNs (40% of SNs and 70% of SNs) and none of SNs use compressive sensing. It can be overserved from all the see from this result that if the number of SNs are increased, the energy consumption in communication (transmission only) is also increased. If compressive sensing is implemented, it reduces the data size which leads to less energy consumption in communication (Figs. 4 and 5).

It can be concluded from above results that data size reduces with the help of compressive sensing and this reduced data consumes less energy for transmission.

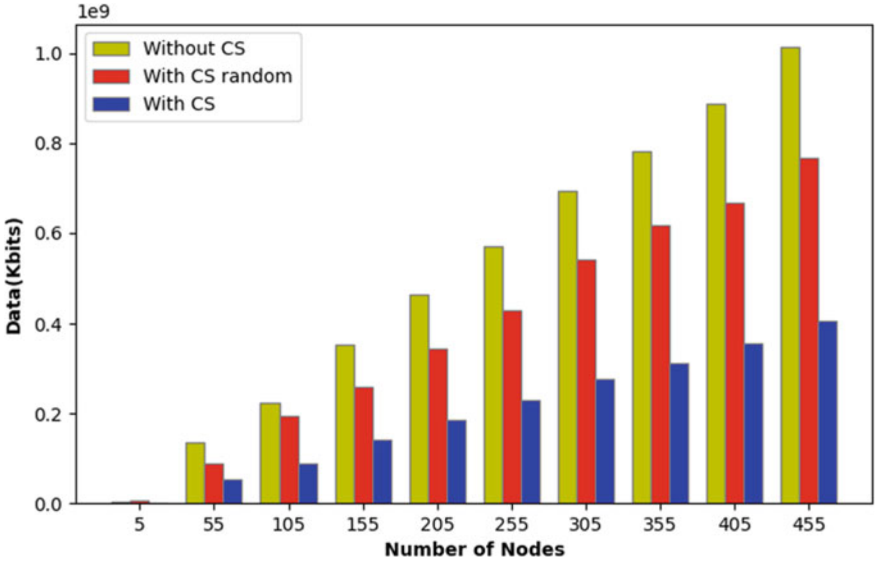


Fig. 2 Data sent by SNs to GNs (40% random CS nodes)

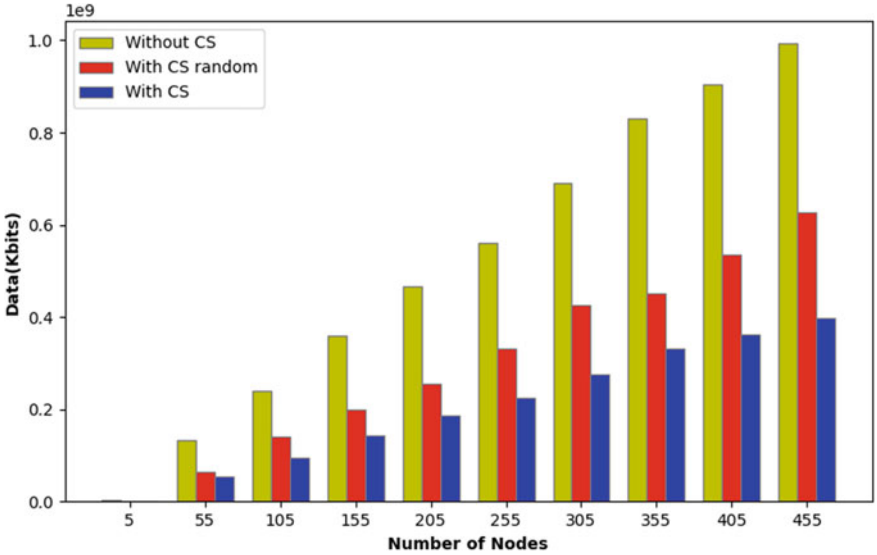


Fig. 3 Data sent by SNs to GNs (70% random CS nodes)

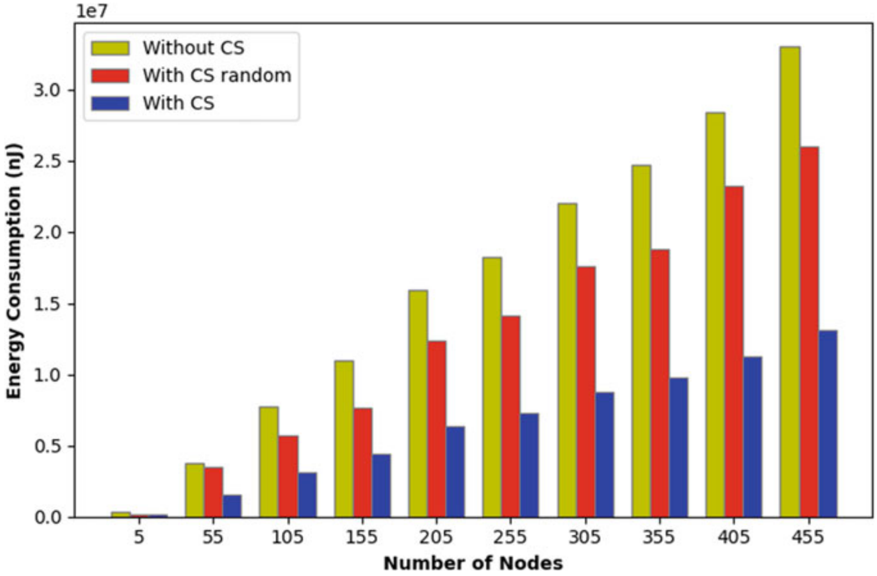


Fig. 4 Energy Consumed by SNs (40% random CS nodes)

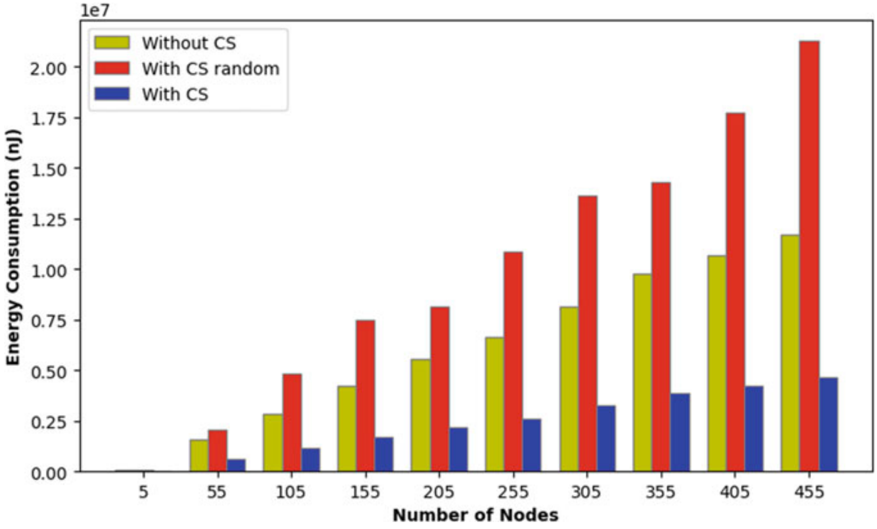


Fig. 5 Energy Consumed by SNs (70% random CS nodes)

Results show that when compressive sensing is used on all SNs, 39.71% energy is consumed. When compressive sensing is used on few randomly selected SNs, 76.32% energy is consumed. The above-mentioned percentage energy consumption is calculated with respect to scenario when no compressive sensing technique is applied

5 Conclusion

The IoT-based systems are divided into three layers in literature and sensing layer has a hierarchical structure of devices with limited battery. In this paper, to save battery compressive sensing is used on SNs which decrease data length and it takes less energy for communication as per Friis energy model. The experiment result shows that a significant amount of energy consumption of SNs can be reduced with compressive sensing by using compressive sensing techniques. If compressive sensing is used on all SNs, 60% of energy is saved for SNs. In this model, only transmission energy is taken into account. For future work, compressive sensing can be considered at GNs and sleep mode for SN and GN nodes. Sensing, processing, and receiving energy as well as reconstruction on BS can also be considered.

References

1. Fuqaha AA, Guizani M, Mohammadi M, Aledhari M, Ayyash M (2015) Internet of things: a survey on enabling technologies protocols and applications. IEEE Commun Surveys Tuts 17(4):2347–2376. 4th Quart
2. Kavre M, Gadekar A, Gadhade Y (2019) Internet of things (IoT): a survey. In: 2019 IEEE pune section international conference (PuneCon), pp 1–6. https://doi.org/10.1109/PuneCon46936.2019.9105831
3. Shah SH, Yaqoob I (2016) A survey: internet of things (IOT) technologies, applications and challenges. In: 2016 IEEE smart energy grid engineering (SEGE), pp 381–385. https://doi.org/10.1109/SEGE.2016.7589556
4. Miorandi D et al (2012) Internet of things: vision, applications and research challenges. J Ad Hoc Netw 10(7):1497–1516
5. Rani S, Talwar R, Malhotra J, Ahmed SH, Sarkar M, Song H (2015) A novel scheme for an energy efficient internet of things based on wireless sensor networks. Sensors 15:28603–28626. https://doi.org/10.3390/s151128603
6. Nižetić S, Šolić P, González-de DLDI, Patrono L (2020) Internet of things (IoT): opportunities, issues and challenges towards a smart and sustainable future. J Clean Prod 274:122877. https://doi.org/10.1016/j.jclepro.2020.122877
7. Kaur N, Sood SK (2017) An energy-efficient architecture for the internet of things (IoT). IEEE Syst J 11(2):796–806
8. Akgul OU, Canberk B. Self-organized things (SoT): an energy efficient next generation network management. Comput Commun. https://doi.org/10.1016/j.comcom.2014.07.004, to be published

9. Zhou Z, Tang J, Zhang L, Ning K, Wang Q (2014) EGF-tree: an energyefficient index tree for facilitating multi-region query aggregation in the Internet of things. Pers Ubiquit Comput 18(4):951–966
10. Tang J, Zhou Z, Niu J, Wang Q (2014) An energy efficient hierarchical clustering index tree for facilitating time-correlated region queries in the Internet of things. J Netw Comput Appl 40:1–11
11. D'Oro S, Galluccio L, Morabito G, Palazzo S (2015) Exploiting object group localization in the Internet of things: a performance analysis. IEEE Trans Veh Technol 64(8):3645–3656
12. Liang J, Chen J, Cheng H, Tseng Y (2013) An energy-efficient sleep scheduling with QoS consideration in 3GPP LTE-advanced networks for Internet of things. IEEE J Emerg Sel Top Circuits Syst 3(1):13–22
13. Wang K, Wang Y, Sun Y, Guo S, Wu J (Dec 2016) Green industrial internet of things architecture: an energy-efficient perspective. IEEE communications magazine-communications standards supplement, pp 48–54
14. Xiang L, Luo J, Rosenberg C (2013) Compressed data aggregation: energy efficient and high-fidelity data collection. IEEE/ACM Trans Networking 21(6):1722–1735
15. Caione C, Brunelli D, Benini L (2012) Distributed compressive sampling for lifetime optimization in dense wireless sensor networks. IEEE Trans Ind Inf 8(1):30–40
16. Luo J, Xiang L, Rosenberg C (2010) Does compressed sensing improve the throughput of wireless sensor networks? In: Processing IEEE ICC
17. Xiang L, Luo J, Vasilakos A (2011) Compressed data aggregation for energy efficient wireless sensor networks. In: Processing 8th annual IEEE communication society conference on sensor, mesh and ad hoc communication and networks
18. Ebrahimi D, Assi C (2013) Optimal and efficient algorithms for projection-based compressive data gathering. IEEE Commun Lett 17(8):1572–1575
19. Ebrahimi D, Assi C (2014) A distributed method for compressive data gathering in wireless sensor networks. IEEE Commun Lett 18(4):624–627
20. Daresari SA, Abouei J (2016) Toward cluster-based weighted compressive data aggregation in wireless sensor networks. Elsevier J Ad Hoc Netw 36:368–385
21. Ebrahim M, Chia WC, Adil SH, Raza K (2019) Block compressive sensing (BCS) based low complexity, energy efficient visual sensor platform with joint multi-phase decoder (JMD). Sensors 19:2309. https://doi.org/10.3390/s19102309
22. Li R, Duan X, Li Y (2019) Measurement structures of image compressive sensing for green internet of things (IoT). Sensors 19:102. https://doi.org/10.3390/s19010102
23. Kuldeep G, Zhang Q (2020) Compressive sensing based multi-class privacy-preserving cloud computing, accepted in IEEE global communications conference
24. Du X, Zhou Z, Zhang Y et al (2020) Energy-efficient sensory data gathering based on compressed sensing in IoT networks. J Cloud Comp 9:19. https://doi.org/10.1186/s13677-020-00166-x
25. Han C, Chen L, Wang W (1 Sept 2021) Utilizing coherent transmission in cooperative compressive sensing in IoT. IEEE Internet Things J 8(17):13555–13566. https://doi.org/10.1109/JIOT.2021.3065829
26. Chen F, Chandrakasan AP, Stojanovic VM (2012) Design and analysis of a hardware-efficient compressed sensing architecture for data compression in wireless sensors. IEEE J Solid-State Circuits 47:744–756
27. Taghouti M, Höweler M (2020) Chapter 22—In-network compressed sensing. In: Fitzek FHP, Granelli F, Seeling P (eds) Computing in communication networks. Academic Press, pp 361–370
28. Djelouat H, Amira A, Bensaali F (2018) Compressive sensing-based IoT applications: a review. J Sens Actuator Netw 7:45. https://doi.org/10.3390/jsan7040045

Dual-Hop Direct Diffusion Routing Protocol for Energy-Efficient Wireless Sensor Network

Sumit Kumar Gupta, Sudhanshu Tyagi, and Sachin Kumar

Abstract In this COVID-19 pandemic situation, health care is on the priority of every human being. The recent development in the miniaturization of intelligent devices has opened many opportunities and played a crucial role in the healthcare industry. The amalgamation of wireless sensor network and Internet of Things is the best example of wireless body area network. These tiny sensor devices have two essential evaluation parameters named as energy efficiency and stability while performing in a group. This paper focuses on various issues of the healthcare system and their solutions. An energy-efficient routing protocol that can provide sensed data to the collection centre or data hub for further processing and treatment of the patients is proposed. Here, we fixed zones for sending data to zone head using distance aware routing, and then zone head send the aggregated data to the data hub. It is better than the low energy adaptive clustering hierarchy (LEACH) by 42% and distance-based residual energy-efficient protocol (DREEP) by 30% in energy efficiency and stability 58% more by LEACH and 39% by DREEP.

Keywords Wireless sensor network · Wireless body area network · Energy-efficient · Direct diffusion · Routing

S. K. Gupta
Department of ECE, Amity School of Engineering and Technology, Lucknow Campus, Lucknow, Uttar Pradesh, India

S. K. Gupta · S. Kumar
Department of ECE, SRMS College of Engineering Technology and Research, AKTU, Lucknow, Uttar Pradesh, India
e-mail: skumar3@lko.amity.edu

S. Tyagi (✉)
Department of Electronics and Communication Engineering, Thapar Institute of Engineering and Technology Deemed to be University, Patiala, Punjab, India
e-mail: s.tyagi@thapar.edu

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_25

1 Introduction

Wireless sensor network (WSN) is prominently used for traditional applications like environment monitoring, military surveillance, civilian surveillance, forest monitoring, agriculture monitoring, etc. However, with the advancement in technology and manufacturing of smaller devices, WSN is now used, from out of the traditional applications, such as precision agriculture [1], irrigation management [2], monitoring of greenhouse gases [3], production process management in agriculture [4], security intrusion detection in the fields [5], wildlife monitoring [6], and many more. So, WSN is not restricted to traditional applications, but it is now more application-specific. So that the end-user can optimize their production, monitoring, efficiency, and be more benefited from their task. As utilization of WSN increased in different fields, researchers also work in the field of health care.

Nowadays, the health monitoring is the most crucial task for human beings. In current scenario, humanity demands the healthcare solutions from academic persons and industry professionals. With the advancement of microelectromechanical systems (MEMS), we are getting tiny sensor devices such as the low-end embedded sensor module [7] which consume few mW of power. WSN is promisingly used in the field of health care. The following monitoring points, such as blood pressure, pulse rate, glucose, body temperature, electrocardiography (ECG), dental care, and others, are general health wellness requirements. A person requires the medical equipment for such monitoring, which is available in the market, to monitor their health. If they are unaware of such equipment and trapped in any medical emergency, they require medical assistance on time. As per the population reference bureau (PRB) report in 2020 [8], the population of age 65 and above are about 6% in India which requires medical treatment and special care by family members. As per the report, there is one doctor for every 1457 people as per the country's current population estimate of 1.35 billion, which is lower than the World Health Organization (WHO) norm of 1:1000, [9]. In this scenario, it is challenging to attend every patient by a doctor. In the healthcare sector, we have to exploit modern technology such as IoT, sensor cloud [1], fog computing [10], and many more as these are the prominent technologies and already explored in different areas.

As the number of patients are growing at a rapid rate and the number of doctors are less, so we have to take the assistance of intelligent devices like sensors, which need to impart in the human body, and using IoT, we can send the data to the respective doctor for analysis and preventive actions. For further analysis, we can store the data in the sensor cloud that other doctors can access for reference in treating other patients. Such patient's data can be monitored and managed by the organization in exchange for money.

1.1 Motivation

The related work has been investigated that a lot of work has been proposed in terms of delay delivery, reliability, metamaterial textile, security, and many more parameters, but energy utilization of the smart sensor devices is uttermost for the sake of wellness of the patient. The motivation of this paper is to provide physical and physiological data, which is compassionate information to concerned doctors or nursing supervisors without delay. For this task, the energy of wireless body area network (WBAN) must be utilized efficiently.

1.2 Contribution

To fulfil the above said motivation, we formulate the problem and propose this paper. Following are the major contributions:

- To increase the energy efficiency of the network, that constituted with intelligent sensor devices, we divide the network into smaller zones.
- Vital information of the patient must receive continuously without delay.
- To increase the stability, we divide the network into four zones with centroid microcontroller zone head (ZH) in each zone.

1.3 Organization

The remaining part of the paper is organized as follows; related work is discussed briefly in Sect. 2. In Sect. 3, we describe the proposed protocol in detail. We simulate the proposed protocol in MATLAB and compare its performance with other established routing protocols in Sect. 4. Finally, we conclude with future scope in Sect. 5.

2 Related Work

IoT paradigm has emerged in a number of different fields in WSN. Researchers are working in health care so that human beings can be benefited from the advancement of technology. In [10], the authors explored the evaluation of the healthcare industry from 1.0 to 4.0 generation. They discussed different challenges in using fog computing in the healthcare 4.0 environment. They provided an analysis of the role of fog computing, cloud computing, and IoT for providing uninterrupted services to the end-users. They introduced a three-layer patient-driven healthcare architecture.

In [11], the authors demonstrated metamaterial clothing that enables wireless signals emitted by intelligent devices to propagate efficiently around the body. They demonstrated that the integration of clothing and metamaterial textile could facilitate physiological signals. Such type of metamaterial is potentially developed for military personnel, sportspersons, and critical patients to continuously monitor health.

In [12], the authors proposed a wearable sensor node powered by solar energy. They have used low energy Bluetooth technology for signal transmission. With the help of such autonomous WBAN, a number of sensor nodes can be implanted to measure body temperature, heartbeat, and other parameters. They also developed a web-based mobile phone application for monitoring sensor data. They have used the MPPT technique to extract maximum power from the solar panel. The proposed work can be extended in terms of wearability and usability.

In [13], the authors proposed an efficient and reliable routing protocol using directional diffusion. They have used the gradient concept for data transmission and minimum hop count for the shortest path. Authors have claimed significantly lesser power consumption and lower packet loss rate in mobile and immobile scenarios.

In [14], the authors proposed certificateless biometric authentication and group key management for validated sensors. Since deployed sensor nodes are restricted in application in a medical environment, effective authentication is required; hence, they have provided security on electrocardiogram (ECG) records as distinctive biometric parameters for authentication procedures.

In [15], the authors proposed an improved certificateless aggregation signature (iCLAS) scheme to protect the sensitive medical information of patients. They have implemented an elliptic curve cryptosystem (ECC), which has an efficient message signing, signature verification, and aggregation algorithm. They have decreased communication and computation costs using complex bilinear pairing operations, which is helpful for the resource-restricted healthcare system.

In [16], the authors proposed a distributed congestion control algorithm for healthcare applications. Due to congestion problems, the vital signs of patients are not collected at the data hub, which deteriorated the reliability and throughput of the healthcare system. They have proposed a priority-based data routing scheme to alleviate the congestion and a priority queue-based scheduling scheme for reliability. Their scheme was used for early alarming in an abnormal sign of patient to state-of-the-art diagnosis.

In [17], the authors proposed a QoS-based, weighted, energy, and temperature-aware routing protocol (WETRP). The route selection scheme uses residual node energy, temperature, and link-delay estimation. The timely delivery of physiological and vital signs of patients in remote monitoring plays a crucial role. The frequent delivery of critical data may be prone to increase collisions and packet loss which impose a detrimental impact on the performance of the healthcare system. In addition to this, intelligent sensor nodes emit electromagnetic radiations, damaging sensitive tissues in the human body. Considering this, the authors have proposed a WETRP scheme for preventing temperature rise, enhancing network lifetime, and reducing the number of hotspots as compared to other state-of-art protocols.

Table 1 A comparison of the-state-of-the-art schemes

Articles	Year	Environment	Energy efficiency	Stability	Security	Reliability
[19]	2015	Heterogeneous	Yes	Yes	–	–
[12]	2017	Heterogeneous	Yes	–	–	–
[11]	2019	Homogeneous	Yes	–	Yes	–
[13]	2019	Homogeneous	Yes	–		Yes
[14]	2019	Homogeneous	Yes	–	Yes	–
[15]	2019	Homogeneous	Yes	–	Yes	–
[17]	2019	Homogeneous	Yes	Yes	–	–
[16]	2020	Heterogeneous	–	–	–	Yes
[18]	2021	Heterogeneous	Yes	Yes	–	–
[20]	2021	Homogeneous	–	–	Yes	–
[21]	2021	Heterogeneous	–	–	Yes	–
Proposed	2021	Heterogeneous	Yes	Yes	–	–

In [18], the authors proposed a segmented sector-based energy-efficient (SSEER) routing protocol for heterogeneous WSN. They divided the network into five sectors, and every sector has normal and advanced sensor nodes. The central sector uses direct diffusion approach with the base station (BS), whereas, other sectors uses clustering mechanism to send the data to the BS. This model can also be used for the healthcare system because of its energy efficiency. The sectors can be considered as disease-specific zones and collected the data to data hub. A systematic comparison of the-state-of-the-art schemes is given in (Table 1).

3 Proposed Approach

In the proposed approach, we adopt the same energy model for healthcare application that was proposed in [22]. As in [23] and [24], direct diffusion protocol is always preferred to small-scale networks as compared to medium and large-scale networks. Therefore, we propose direct diffusion with two-hop transmissions from the sensor device to the data hub. In LEACH [22], clusters are formed based on distance nearer to cluster head (CH). These CHs aggregate the data they receive from cluster members (CMs) and then transmit the data to the base station (BS). Nevertheless, in the proposed model, network is divided into four zones, and every zone has its microcontroller ZH to receive the data. All ZHs aggregate the data and transmit the aggregated data to the data hub, which is placed at the centre of the network. The proposed network model is shown in Fig. 1.

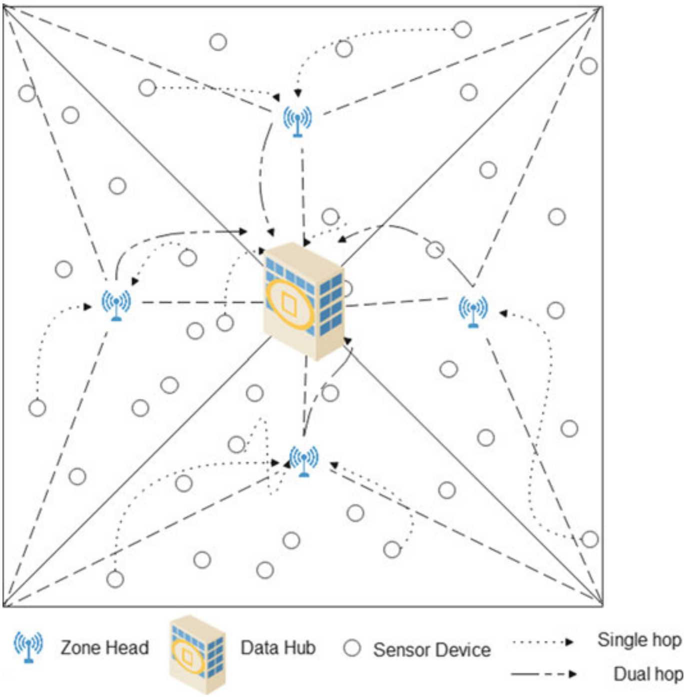


Fig. 1 Network model

3.1 *Zone Selection*

We design the proposed model for a healthcare system that must be energy-efficient so that we can fix the zone size as per the requirement. In the proposed model, we use a square network and the data hub placed at the centre of the network. Network is divided into triangular shape zone. Four sides of the square network consider as the base of the triangular zones, and the other two sides of the triangular zone meet at the centre of the network. The selection of zones is based on the following assumptions:

- All zones are fixed, like hospitals and residential buildings.
- All the patients are considered as an immovable or negligible movement.
- Sensor devices that are intact with patients have limited power.

The deployment of sensor devices is based on the boundary between two zones explained in algorithm1.

Algorithm 1 Pseudocode for Zone Selection

```
Input: (x, y) co-ordinates ∈ sensor devices, boundary equation, r ∈ number of devices in zone
    in
Output: $zone_{selection}$ out
Initialisation : Start
find (x, y) co-ordinates ∈ sensor device,
LOOP process for Zone
for $i = l$ to $r$ do
    if $(x, y) inequation > 0$ then
        Upper Zone
    else
        Lower zone
    end if
end for
return $Start$
```

3.2 Zone Head Selection

In the proposed model, energy consumption is reduced in the cluster formation, which was assumed zero in [22] but practically, hardly possible. To overcome this assumption and to receive data continuously at the data hub, we use direct diffusion using single-hop and dual-hop communication in the clusters. We fix ZH on the centroid of the triangular zone so that maximum sensor devices can send vital information to the minimum distance, and low energy consumption can be achieved during the information transmission.

Suppose the triangular zone has three vertexes (x_1, y_1), (x_2, y_2), and (x_3, y_3), respectively. So, the centroid can be mathematically expressed as in Eq. (1):

$$G(x, y) = ((x_1 + x_2 + x_3)/3, (y_1 + y_2 + y_3)/3) \tag{1}$$

All the ZHs fix at their centroid position in their respective zones are shown in Fig. 2.

3.3 Execution of the Proposed Protocol

In the proposed protocol, network is divided into four zones, and every zone has its ZH on the centroid position since all the sensor devices fix at a geographically known location so their distance from ZH and data hub can be easily calculated. The proposed protocol uses single-hop or dual-hop communication based on distance calculation from ZH and data hub.

All the sensor devices send the data to ZH if their distance to the data hub is greater than the distance to ZH. It is dual-hop communication because the transmitted data first goes to ZH and then, after aggregation, it goes to the data hub.

All the sensor devices send the data to the data hub if their distance to the data hub is lesser than to ZH. It is single-hop communication because the transmitted data directly goes to the data hub. Detailed execution of our proposed scheme is presented in algorithm 2.

Algorithm 2 Pseudocode for Proposed scheme

```
Input: distance, m $\in zone_{device}$ in
Output: hop out
Initialisation : Start
calculate the distance of sensor device from ZH and data hub
LOOP process for zone-wise
for $i = l$ to $m$ do
    if $dis_{ZH} < dis_{datahub}$ then
        Dual hop communication
    else
        Single hop communication
    end if
end for
return Start
```

4 Performance Evaluation

In this section, we discuss the performance of the proposed model with the state-of-the-art routing protocols like LEACH [22], DREEP [19], SEP [25], EHE-LEACH [24], and SSEER [18]. The performance metrics are network lifetime and stability, measured as last node alive (LNA), and first node die (FND), respectively. Network lifetime reveals as how many number of rounds can sensor device send data to the data hub, while stability reveals as how many rounds, first sensor device dies. Half node alive (HNA) is, only 50% remaining, alive sensor devices. MATLAB is the platform in which all the simulations are performed. There are 100 sensor devices and four microzone controllers as ZHs. These ZHs have α times higher energy than normal sensor devices because they also sense the information along with aggregation operation. All sensor devices are stationary. Further, data hub is fixed at the centre of network, i.e. 50 m × 50 m and there is no shortage of power. Table 2 gives the details of various parameters used in the simulation.

Two network scenarios are used, i.e. the first one is homogeneous and the second one is heterogeneous, to compare the proposed protocol with other state-of-the-art routing protocols.

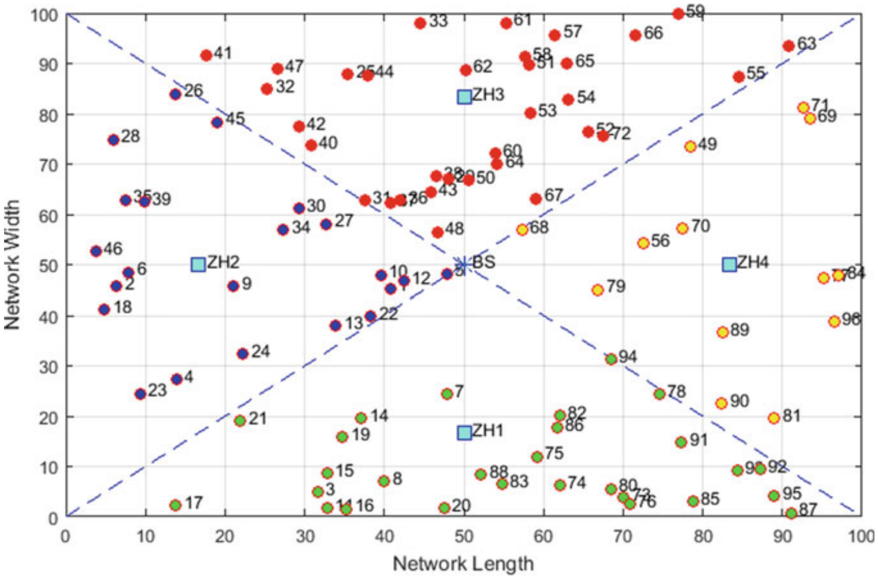


Fig. 2 Zone-wise sensors deployment

Table 2 Simulation parameters

Parameter	Value
Amplifier energy for multipath (E_{mp})	0.0013×10^{-12} J/bit/m^4
Amplifier energy for free space (E_{fs})	10×10^{-12} J/bit/m^2
Amplifying factor (α)	2
Data aggregation energy (EDA)	5×10^{-9} J/bit/round
Energy of each sensor device (E_n)	0.25 J
Heteroginity factor of devices	10% of N
Network size	100 × 100
Packet size (p)	4000 bits
Receiver circuitry dissipation (E_{Rx})	50×10^{-9} J/bit
Transmitter circuitry dissipation (E_{Tx})	50×10^{-9} J/bit
Total sensor devices (N)	100
ZH energy (E_{ZH})	$\alpha\ E_n$

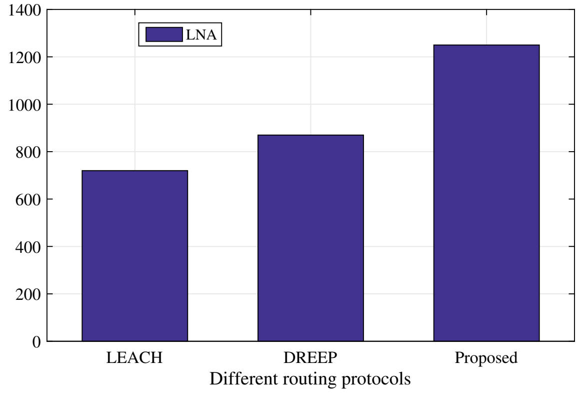


Fig. 3 Lifetime of routing protocol in scenario-1

4.1 Scenario-1: Homogeneous Environment

In the homogeneous environment, all the sensor devices have equal energy (E_n). We investigate network lifetime as LNA and stability as FND of the proposed protocol with the-state-of-the-art routing protocols LEACH [22] and DREEP [19]. During the investigation, all the environment parameters keep same for all comparing protocols.

Figure 3 shows the network lifetime of the proposed protocol that enhances significantly as compared to [22] by 42% and [19] by 30%. Figure 4 shows the stability of the proposed protocol that enhances significantly as compared to [22] by 58% and [19] by 39%. This enhancement is due to less communication distance and no other overheads like message acknowledgement with other sensor devices.

WBAN is the network of different implanted sensors in the patients. Patient wellness is a paramount and fundamental concern in the healthcare system. The early detection of any fluctuation in physiological and vital signs can lead to the safety of patients. If we get the data continually, it is not essentially effective in some vital signs such as temperature and pulse rate and physiological data such as blood pressure and skin conductance in non-critical conditions. Such cases may also lead to deterioration of patients health due to radiation of electromagnetic waves, as discussed in [17].

Considering this fact, we investigate the energy efficiency and stability of our system for real time and event-driven applications. In an event-driven application, no need to send the data continually to the monitoring team. Hence, we modify the system in a threshold-based system. Such a system has some critical or threshold value. If the measured information is below that value, no need to send information to the data hub, and if it is above that value, send the information immediately with an alarming message. In this way, we can reduce the energy consumption of sensor devices. In Fig. 5, the difference between real time and event-driven energy-efficient systems can be observed.

Fig. 4 Stability of routing protocol in scenario-1

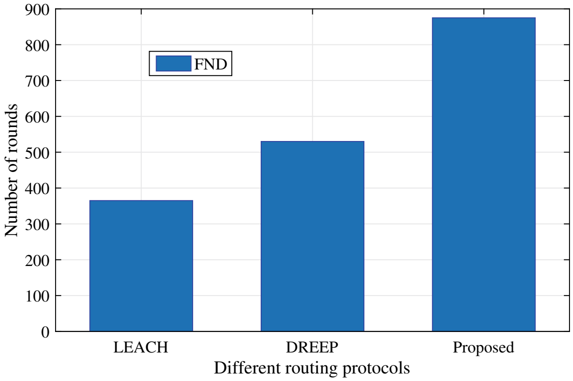
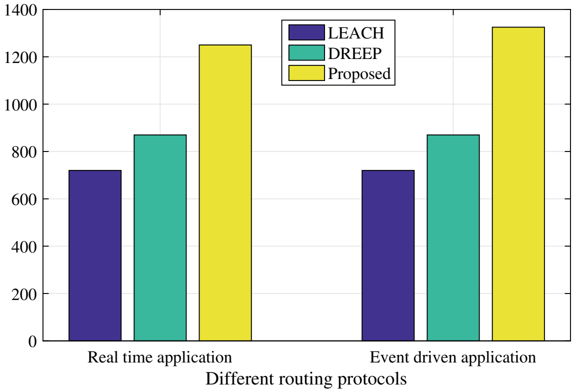


Fig. 5 Comparison of lifetime between real time and event-driven application



4.2 Scenario-2: Heterogeneous Environment

In the heterogeneous environment, only 10% of the total devices have α times higher energy of each device.

As shown in Fig. 6, network stability is around 56% higher as compared to SEP and 52% as compared to EHE-LEACH, whereas 32% higher than SSEER in terms of FND because of the lesser distance between device and ZH. They are not involved in any transmission loss in finding CH in every round. Similarly, Fig. 7 shows 47% significant improvement in HNA in proposed protocol as compared to SEP, 42% in EHE-LEACH, and 23% in SSEER. All these improvements are in the real-time application in which devices are sending data continually to ZH or data hub.

Fig. 6 Stability of routing protocol in scenario-2

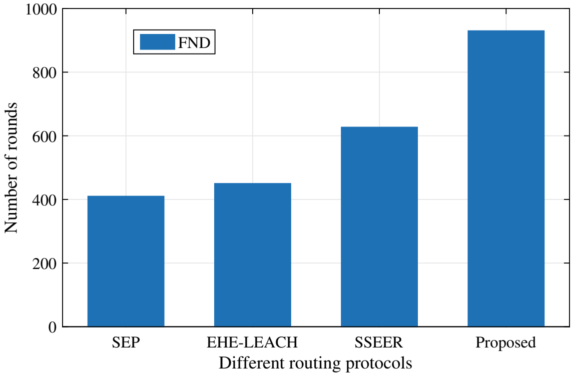


Fig. 7 Lifetime of routing protocol in scenario-2

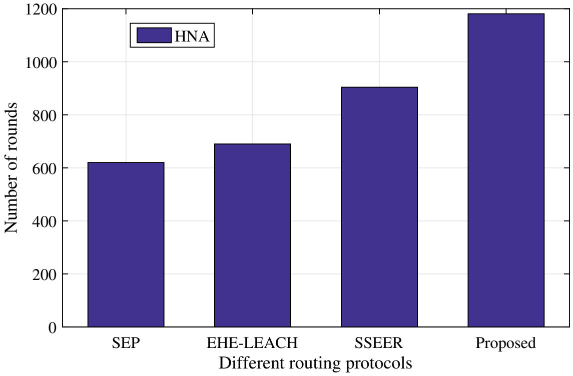
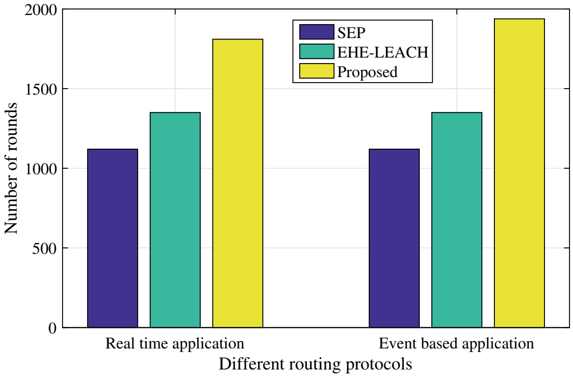


Fig. 8 Comparison of lifetime between real time and event-driven application



In event-driven application continuous data transmission to the monitoring team is not required, so, longer lifetime of the system is achieved, as shown in Fig. 8. Since data is transmitted only if an emergency alarms or vital sign shows an abrupt change in threshold value.

5 Conclusion

In this paper, we propose energy-efficient and stable routing protocols in different environments using different performance metrics. Simulations performed in a homogeneous and heterogeneous environment because patients suffer from different indispositions in hospitals. In this model, microcontroller zone heads are used for sensing and aggregation of data that receive from different sensor devices in its zone. Further, single or dual-hop for data transmission is selected zone-wise that enhance the system energy, which can save more power for the sensor device. Simulation shows that the proposed scheme is better than LEACH by 42% and DREEP by 30% in energy efficiency. It shows stability 58% and 39% higher than LEACH and DREEP, respectively, in a homogeneous environment. It shows stability 56%, 52%, and 32% higher than SEP, EHE-LEACH, and SSEER in a heterogeneous environment. For future scope, we can further divide the zones into microzones for specific diseases so that we can collect the data efficiently from microzones.

References

1. Ojha T, Misra S, Raghuwanshi NS (2017) Sensing-cloud: leveraging the benefits for agricultural applications. Comput Electron Agric 135:96–107
2. Nikolidakis SA, Kandris D, Vergados DD, Douligeris C (2015) Energy efficient automated control of irrigation in agriculture by using wireless sensor networks. Comput Electron Agric 113:154–163
3. Malaver A, Motta N, Corke P, Gonzalez F (2015) Development and integration of a solar powered unmanned aerial vehicle and a wireless sensor network to monitor greenhouse gases. Sensors 15:4072–4096 Feb
4. Dong X, Vuran MC, Irmak S (2013) Autonomous precision agriculture through integration of wireless underground sensor networks with center pivot irrigation systems. Ad Hoc Networks 11:1975–1987
5. Roy SK, Roy A, Misra S, Raghuwanshi NS, Obaidat MS (2015, June) AID: a prototype for agricultural intrusion detection using wireless sensor network. In: 2015 IEEE international conference on communications (ICC), London, UK. IEEE, New York
6. Dominguez-Morales JP, Rios-Navarro A, Dominguez-Morales M, Tapiador-Morales R, Gutierrez-Galan D, Cascado-Caballero D, Jimenez-Fernandez A, Linares-Barranco A (2016) Wireless sensor network for wildlife tracking and behavior classification of animals in Donana. IEEE Commun Lett 20:2534–2537
7. Gupta SK, Kumar S, Tyagi S, Tanwar S (2020) Energy efficient routing protocols for wireless sensor network. In: Advances in intelligent systems and computing. Springer International Publishing, pp 275–298
8. PRB. Population Reference Bureau (2021) https://www.prb.org/international/geography/india
9. PTI. Business-Standard (2019) https://www.business-standard.com/article/pti-stories/doctor-patient-ratio-in-india-less-than-who-prescribed-norm-of-1-1000-govt-119111901421_1.html
10. Kumari A, Tanwar S, Tyagi S, Kumar N (2018) Fog computing for healthcare 4.0 environment: opportunities and challenges. Comput Electr Eng 72:1–13
11. Tian X, Yang X, Ho JS, (2019, Oct) Energy-efficient and secure wireless body sensor networks with metamaterial textiles. In: IEEE biomedical circuits and systems conference (BioCAS), Nara. Japan, IEEE, New York

12. Wu T, Wu F, Redoute J-M, Yuce MR (2017) An autonomous wireless body area network implementation towards IoT connected healthcare applications. IEEE Access 5:11413–11422
13. Mu J, Yi X, Liu X, Han L (2019) An efficient and reliable directed diffusion routing protocol in wireless body area networks. IEEE Access 7:58883–58892
14. Tan H, Chung I (2019) Secure authentication and group key distribution scheme for WBANs based on smartphone ECG sensor. IEEE Access 7:151459–151474
15. Xie Y, Li X, Zhang S, Li Y (2019) An improved certificateless aggregate signature scheme for healthcare wireless sensor networks. IEEE Access 7:15170–15182
16. Chanak P, Banerjee I (2020) Congestion free routing mechanism for IoT-enabled wireless sensor networks for smart healthcare applications. IEEE Trans Consumer Electron 66:223–232
17. Bhangwar AR, Ahmed A, Khan UA, Saba T, Almustafa K, Haseeb K, Islam N (2019) WETRP: weight based energy & temperature aware routing protocol for wireless body sensor networks. IEEE Access 7:87987–87995 June
18. Gupta SK, Kumar S, Tyagi S (2021, July) SSEER: segmented sectors in energy efficient routing for wireless sensor network. Multimedia Tools Appl (Accepted for Publication)
19. Mittal N, Singh U (2015) Distance-based residual energy-efficient stable election protocol for WSNs. Arab Jo Sci Eng 40(6):1637–1646
20. Bhavin M, Tanwar S, Sharma N, Tyagi S, Kumar N (2021) Blockchain and quantum blind signature-based hybrid scheme for healthcare 5.0 applications. J Inf Secur Appl 56:102673
21. Bhattacharya P, Tanwar S, Bodkhe U, Tyagi S, Kumar N (2021) BinDaaS: blockchain-based deep-learning as-a-service in healthcare 4.0 applications. IEEE Trans Netw Sci Eng 8:1242–1255
22. Rabiner HW, Anantha C, Hari B (2000, Jan) Energy-efficient communication protocol for wireless microsensor networks. In: Proceedings of the 33rd annual Hawaii international conference on system sciences, vol 2, (Maui, HI, USA). IEEE Computer Society, pp 1–10
23. Tyagi S, Kumar N (2013) A systematic review on clustering and routing techniques based upon LEACH protocol for wireless sensor networks. J Network Comput Appl 36:623–645
24. Tyagi S, Gupta SK, Tanwar S, Kumar N (2013, Aug) EHE-LEACH: enhanced heterogeneous LEACH protocol for lifetime enhancement of wireless SNs. In: 2013 international conference on advances in computing, communications and informatics (ICACCI), (Mysore, India), pp 1485–1490
25. Smaragdakis G, Matta I, Bestavros A (2004, May) SEP: a stable election protocol for clustered heterogeneous wireless sensor networks, pp 1–11, OpenBU

Design and Development of Smart Waste Bin for Effective Waste Collection and Management

Abidemi M. Orimogunje, Olamide V. Fred-Ahmadu, Adeyinka A. Adewale, Alashiri Olaitan, Sanjay Misra, Akshat Agrawal, and Ravin Ahuja

Abstract In modern cities, waste management has become a problem as waste is being generated faster than it is being managed, and this is leading to serious environmental pollution. Most conventional trash bins in place are not being emptied efficiently enough which leads to the overflow of waste and the breed of pests, insects and the dreadful diseases they cause and spread. It is a known fact that carbon dioxide and other air pollutant resulting from unmanaged wastes causes global warming which is dangerous to human wellbeing. So, this work proposes a smart waste bin to curtail indiscriminate disposal of waste and it incorporate the use of mobile technology to notify the waste management personnel for timely and effective handling of the waste. The designed intelligent waste bin is made up of two systems. The automated lid system is the first part, and it is responsible for the opening and closing the cover-lid of the bin without the need for physical contact between the user and the bin. It also prevents the user from coming in contact with the waste-borne germs, bacteria, and viruses that may be present on the surface of the waste bin. The second is the communication system which will be responsible for informing the waste management authorities about the level of the waste in the bin so that the bin can be emptied as soon as possible. The smart bin opens when the incoming trash is at a distance of 20 cm (or less) to the bin. The bin will not open at a distance greater than this threshold of 20 cm, likewise it will not open when the waste bin is 95% full relative to its capacity. The waste management personnel are notified through

A. M. Orimogunje · O. V. Fred-Ahmadu · A. A. Adewale · A. Olaitan
Center of ICT/ICE Research, Covenant University, Ota, Ogun, Nigeria
e-mail: abidemi.orimogunje@covenantuniversity.edu.ng

S. Misra
Department of Computer Science and Communication, Østfold University College, Halden, Norway
e-mail: sanjay.misra@hiof.no

A. Agrawal (✉)
Amity University, Gurgaon, Hariyana, India
e-mail: akshatag20@gmail.com

R. Ahuja
Shri ViswakarmaSkill University, Gurgaon, Hariyana, India

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_26

the communication system attached to the smart bin when the bin is 95% full so that the waste in the bin could be emptied. This prevents trash overflow from the bin and ultimately makes the environment cleaner, more eco-friendly and prevent global warming.

Keywords Smart bin · Arduino Nano · Ultrasonic sensor · Servo motor · GSM module

1 Introduction

In the last few years, urbanization, population and industrialization has increased at a tremendous rate and with it, the rate of waste production has increased as well [1–3]. The overflow of waste leads to environmental pollution which causes the area affected to look unsightly [4]. The foul odor of the open waste makes the region unconducive for those who live nearby. All of this can lead to a repulsive and unhygienic environment that allows for the breeding of pests, pathogens, and insects that spread all kinds of terrible diseases [5]. The current global situation around the COVID-19 epidemic, as well as the modifications and restrictions that come with it, has necessitated the need to minimize physical contact with people and objects particularly with dirty surfaces[6]. In most cases, people dispose waste indiscriminately because of the need to avoid contact with dirty lid of such waste bin. Aside the possibility of flooding due to the blockage of drainage by waste products, air pollution and diseases could result from accumulated wastes [7]. These general environmental pollution caused by indiscriminate waste disposal are all symptoms of poor waste management, which is why the smart waste bin is being developed to improve waste collection and management [8, 9]

This project proposes a smart trash bin as the solution to coming in contact with trash bin lids. Figure 1 shows the processes involved in the operation of the smart bin.

The smart bin will automatically open upon sensing oncoming trash and will shut its lid after a few seconds. This takes away the need for contact with the waste bin, thereby eliminating the risk of contracting germs and bacteria capable of spreading disease. This also ensures the lid remains closed when not in use which ultimately helps to contain the smell of the trash within it. Also, the smart bin will not be opened to receiving trash when the bin is full, as a result, overflow will be prevented, and an SMS text message will be sent to the appropriate authorities via the GSM module incorporated in the design, alerting them that the bin is at full capacity and should be emptied as soon as possible.

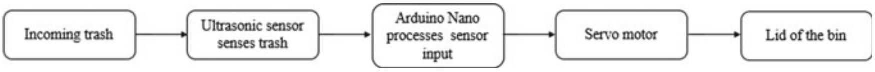

Incoming trash
Ultrasonic sensor senses trash
Arduino Nano processes sensor input
Servo motor
Lid of the bin

Fig. 1 Operation of the smart bin

Waste management will become more effective as smart trash bins are designed and implemented, and cities will become cleaner, healthier, and more environmentally friendly, which is a positive step toward achieving sustainable smart cities [8]. This work seeks to contribute the following:

- Prevent indiscriminate disposal of waste
- Curtail the spread of germs and bacteria
- Effectively manage waste disposal through intelligent and communication system
- Minimize global warming through timely handling of the waste which could lead to emission of Greenhouse gases.

The remaining section of the article is as follows: Sect. 2 captures the previous work that has been done relating to this project and the social relevance of the project. Section 3 is about the components used in the project and its relevance to achieving a smart waste bin, Sect. 4 explains the principle of operation of the project with a flow chart describing the flow of operation, while the conclusion and the future recommendation is given in Sect. 5.

2 Related Works

Sathyakala et al. in [10] proposes a smart trash bin system that would ensure garbage collection is done only when needed to reduce garbage collection costs and prevent waste overflow. The proposed system uses an infrared sensor to measure the waste level of the bin and a PIC microcontroller as an interface through which the hardware is programmed. The PIC microcontroller is a powerful microcontroller but the Arduino Nano is just as powerful, efficient, cost-effective, and much easier to use and program.

Sohag and Podder [1] proposes a smart bin which is similar to the one developed in this project. However, an Arduino Uno board was used instead of the Arduino Nano board used in this work. Though the two microcontroller offers the same functionalities, but the Nano board used in this work is smaller and cheaper.

Bhatt et al. [11] proposed a smart bin that uses an ultrasonic sensor to detect oncoming trash. Based on the sensors reading and the predefined parameters coded into the microcontroller, the servo motor placed on the lid of the bin opens it for the trash to be collected and then closes it automatically after a few seconds. The bin uses a less efficient microcontroller instead of the Arduino Nano microcontroller used in this work.

Sreejit et al. [12] proposed a smart bin that uses a PIC board for a microcontroller, an ultrasonic sensor for waste level detection and a GSM module for its communication system. This system also has an android compatible mobile application on the client end. Through the application, the user can locate the nearest available bin to use and the fastest route to it.

In [13] the author gave a review on the importance of big data to the Internet of vehicles and smart cities. The impact of the big data to the safety and security of environment. The data helps in taking decision about urban planning and environmental safety. Though this project does not involve big data, yet the rate at which the bin get filled up could be used in the management of the smart bin and proper planning.

Authors in [14] develop a rain flood ecosystem for urban cities where climate change and global warming are controlled. The work was basically on mitigating the risk of flood in smart cities. The authors failed to give credence to the role of proper waste management in ensuring effective and timely handling of wastes that could lead to the blockage of drainage which causes flooding in some cases in smart cities. Researches are going on various issues, and challenges on smart city management [15–17].

- **Social Relevance**

In urban cities, most conventional trash bins in place are not being emptied efficiently enough and this leads to the overflow of waste and the breed of pests, insects, and the dreadful diseases they cause and spread.

The conventional trash bins require physical contact to use, this is no longer ideal especially at this period of global pandemic due to the COVID-19 outbreak, which necessitated minimum or no physical contact with people or object. Therefore, this project will further prevent the spread of diseases.

3 Methodology

The major components used in this project are an Arduino Nano microcontroller board, two ultrasonic sensors, a servo motor, a GSM module (SIM800L), and a liquid crystal display (LCD).

Other materials like jumper wires, Vero board, hot glue, and so on were crucial to the achievement of the project.

The system is in two parts which are;

i. The automated trash collection system,
ii. The bin communication system.

The two systems are connected to the same Arduino Nano board. The automated trash collection system involves an ultrasonic sensor (HC-SR04) that uses a technique similar to echolocation, which bats use to detect nearby objects, to detect oncoming trash as shown in Fig. 2.

The ultrasonic sensors measure the distance between the incoming trash and the bin, then promptly feeds the received data (in form of measured distance) to the Arduino Nano. The Arduino Nano then compares the distance measured by the sensor to the set threshold of 20 cm that has been programmed into it. If the distance measured is less than or equal to 20 cm, the Arduino Nano energize the servo motor to open the lid but if the distance between the bin and the object is greater than 20 cm, the bin stays shut as shown in the Fig. 2. The system was programmed with such a

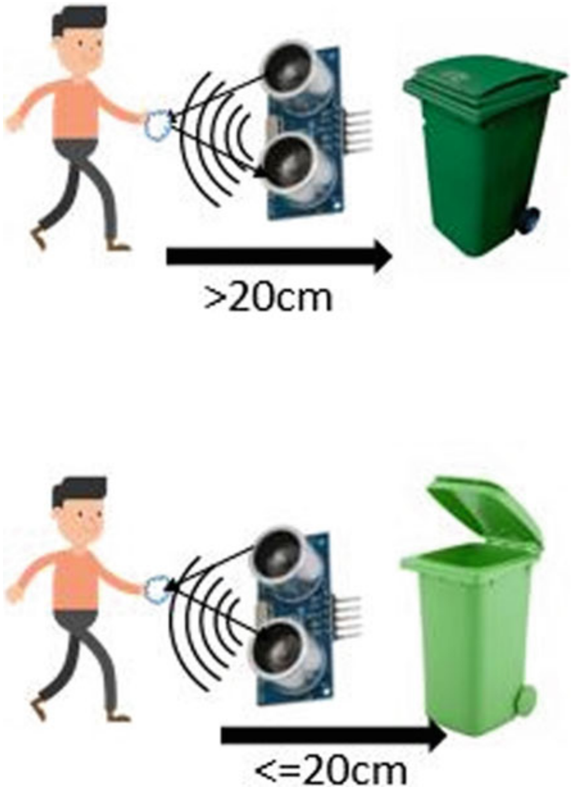


Fig. 2 Automatic opening of the bin lid

small threshold distance to prevent the bin from opening unnecessarily when people pass by it. It is expected that the user would walk up to the bin and extend the trash toward it. The smart waste management system involves another ultrasonic sensor present inside the bin that measures the height of the trash in the bin. With this sensor present, the capacity of the bin can be measured and displayed on the liquid crystal display (LCD) in real time.

4 Implementations

As soon as the trash level in the bin measures up to 70%, a text message will be sent to the appropriate personnel telling them the bin is almost full. However, the bin will continue to collect more trash but when the trash level measures up to 95%, another message will be sent to the appropriate personnel, telling them that the bin is full. At this point, the bin shuts down and stops opening to collect more trash till the trash collection personnel arrive to empty it. Once the bin is empty, the process starts again and the cycle continues until the waste in the bin is 95% of its capacity, then the lid of the bin refuse to open for more trash to come in. This is depicted in the flowchart in Fig. 3.

For the automatic lid system of the bin, the Arduino Nano, ultrasonic sensor, and servo motor work together.

The Arduino Nano is a microcontroller board that is quite synonymous to the Arduino Uno but is significantly smaller than the Uno in size, making it all the more suitable for embedded system applications. It is called a microcontroller because it has input and output pins, a processing unit which is the ATmega328P and memory

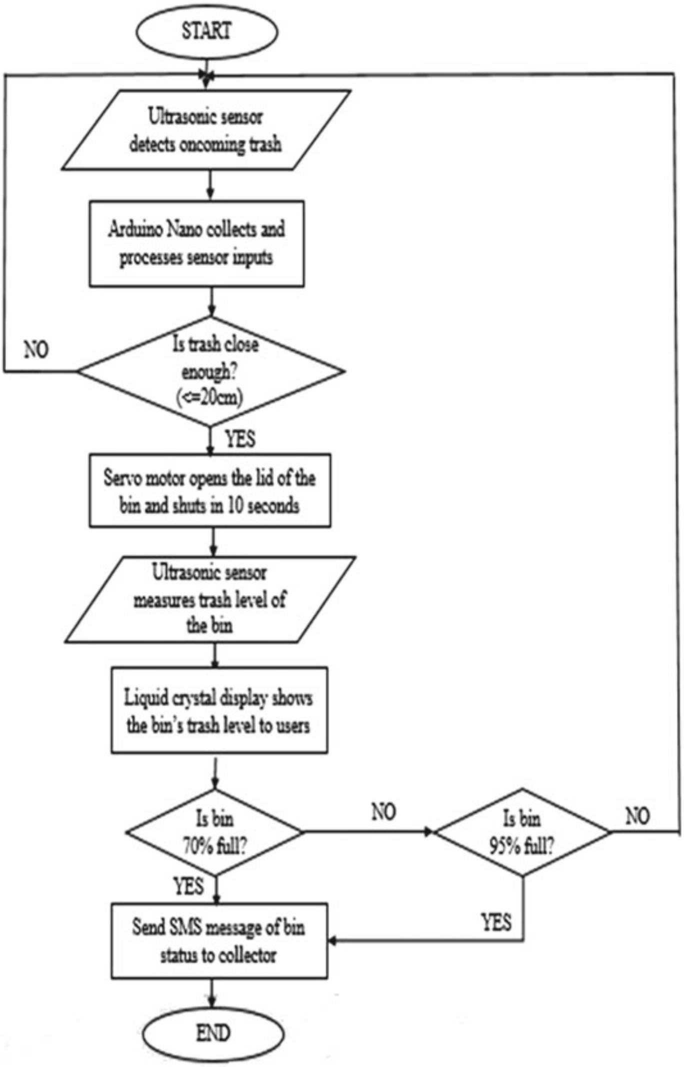

START
Ultrasonic sensor detects oncoming trash
Arduino Nano collects and processes sensor inputs
Is trash close enough? (<=20cm)
NO
YES
Servo motor opens the lid of the bin and shuts in 10 seconds
Ultrasonic sensor measures trash level of the bin
Liquid crystal display shows the bin's trash level to users
Is bin 70% full?
NO
Is bin 95% full?
NO
YES
YES
Send SMS message of bin status to collector
END

Fig. 3 Flowchart of the smart bin system

for the storage of instructions. Regarding this project, the Arduino Nano is the brain of the circuit. Being the microcontroller in the circuit, it is responsible for instructing all the other components, telling them what to do and when to act. Therefore, all the other components of the circuit are connected to its pins via jumper wires. The Arduino Nano board has memory, where it stores all the instructions for the components connected to it. These instructions are written in the Arduino IDE using the C++ programming language and then sent to the memory in the Arduino.

The ultrasonic sensor measures distance with the use of ultrasonic waves. When the sound bounces off the target object and is reflected back to the sensor, it deduces the distance between the target object and itself by measuring the time that the transmission and reflection took. The following formula can be used to calculate the distance:

$$\text{Distance } L = 1/2 \times \mathrm{T} \times \mathrm{C} \tag{1}$$

where L is the desired distance, T is the time between the ultrasonic wave emission, and reception and C is the sonic speed. (The value is multiplied by 1/2 because T is the round-trip time (go-and-return) of ultrasonic waves).

The Vcc pin is connected to the 5v pin on the Arduino Nano so that the sensor can draw power. The ground pin is connected to the ground on the Arduino Nano. The trigger pin is responsible for the emission of the sound waves that are expected to travel from the sensor, bounce off the target object and reflect to the sensor so the desired distance can be measured. When the sound waves are reflected back to the sensor, the Echo pin goes high for a set time, which is equal to the time it takes for the wave to return to the sensor. This time is measured and used in calculating the distance between the object and the sensor. The trigger pin and the echo pin are connected to two of the digital pins on the Arduino Nano as they both deal with transmission and reception respectively.

A servo motor is a self-contained electrical device that efficiently and precisely rotates machine parts [18]. It could be either a linear or rotary actuator which gives precise position control in closed-loop applications.

Servo motors typically have three wires, namely, power (red), ground (brown), and signal (yellow). The ground pin is connected to ground, the power pin is connected to the power plane and the signal pin is connected to a digital pin on the Arduino.

These components and the connections of the different components causes the automatic opening and closing of the bin.

For the bin communication system, the GSM module (SIM800L), ultrasonic sensor, and the Arduino Nano work together to establish communication between the bin and the garbage collection personnel as shown in Fig. 4.

The liquid crystal display shows the user the level of trash in the bin so it is obvious if the bin can take more trash or if it is full.

The SIM800L is a tiny GSM/GPRS module that supports SIMCOM enhanced AT commands and provides 2G GSM/GPRS data. It is simple to interface with the UART of almost all popular microcontrollers because it uses the serial communication method. It has four pins connecting it to the Arduino Nano. The Vcc pin is connected

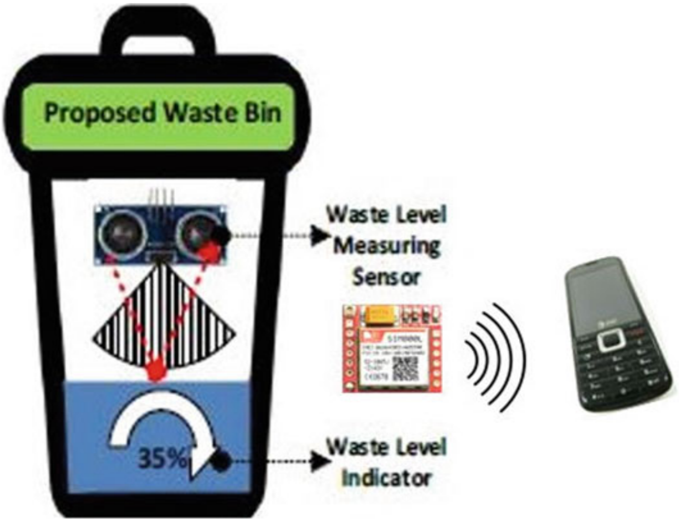


Fig. 4 Bin communication system

to the 5v pin on the Arduino, the ground pin is connected to ground, the TX and RX pins, which are responsible for the transmission and reception of signal, are connected to two of the digital I/O pins on the Arduino.

The ultrasonic sensor is placed inside the bin, under the lid. The Vcc pin is connected to the 5v pin on the Arduino Nano so that the sensor can draw power. The ground pin is connected to the ground on the Arduino Nano. The trigger pin and the echo pin are connected to two of the digital pins on the Arduino (what are the purpose). The resulting design is shown in the Fig. 5

The Fig. 5 shows the final combination of the different components which result in the developed smart bin.

5 Conclusion and Future Work/Recommendation

The smart bin is crucial in the achievement of smart cities which is one of the targets of the United Nations Sustainable development goals and this will lead to a clean and sustainable environment. It can be used for both outdoor and indoor applications. The project will eliminate the need for physical contact with trash bins and improve waste management systems and the logistics involved, thereby reducing environmental pollution, the spread of waste borne diseases and pest infestation and boosting general public health.

In the future, the work can be enhanced with the following:

- Use of solar panel systems to power the smart waste bin

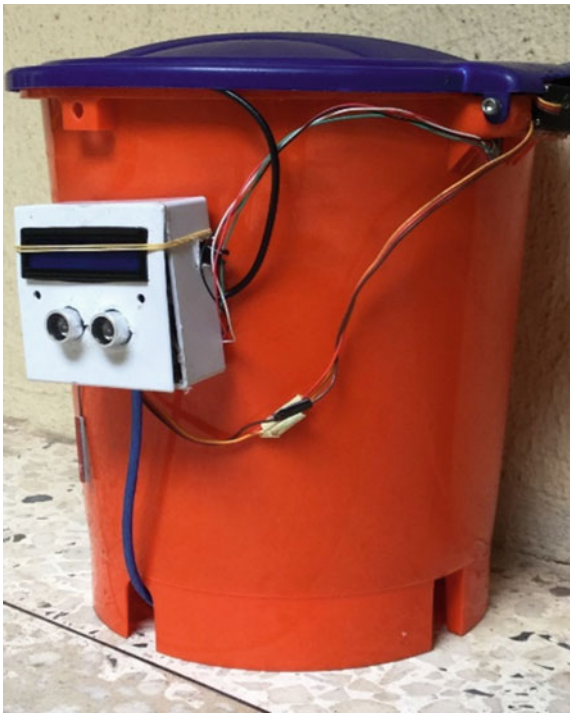


Fig. 5 The designed smart waste bin

- Incorporation of the Internet of Things to help users know the location of the nearest waste bin and the different waste bin available for disposal of different form of waste.
- Adding a compressor function to the bin so it can contain as much trash as possible per system cycle. Trash like nylon bags and paper tend to take space in the bin and make it seem full when they can simply be compressed, allowing the bin take more trash. The compressor will prevent the bin from filling faster than it should.

References

1. Sohag MU, Podder AK (2020) Smart garbage management system for a sustainable urban life: an IoT based application. Int Things 11:100255. https://doi.org/10.1016/j.iot.2020.100255
2. Narayanan M (2017) Smart garbage monitoring system using sensors with RFID over internet of things. J Adv Res Dyn Control Syst 9(7):133–140
3. Yan Y, Zhang Y, Sharma A, Al-Amri JF (2021) Evaluation of suitability of urban land using GIS technology. Sustain 13(19). https://doi.org/10.3390/su131910521
4. Bin S (2017) Waste management u sing solar. 2017 International conference energy, communication data analysis soft computer no. Dc, pp 1123–1126
5. Zavare S, Parashare R, Patil S, Rathod P, Babanne V (2017) Smart city waste management system using GSM. Int J Eng Sci Comput 5(3):74–78

6. Santos C, Penteado G, Aurélio M, De Castro S (2020/2021) Resources, conservation & recycling Covid-19 effects on municipal solid waste management: what can effectively be done in the Brazilian scenario ? Resour Conserv Recycl 164(8):105152. https://doi.org/10.1016/j.resconrec.2020.105152
7. Sinha A, Gupta K, Jamshed A, Singh RK (2020) Intelligent dustbin: a strategic plan for smart cities. Mater Today Proc. https://doi.org/10.1016/j.matpr.2020.09.529
8. Lu X, Pu X, Han X (2020) Sustainable smart waste classification and collection system: a bi-objective modeling and optimization approach. J Cleaner Prod 276. https://doi.org/10.1016/j.jclepro.2020.124183
9. Ashwin M, Alqahtani AS, Mubarakali A (2021) IoT based intelligent route selection of wastage segregation for smart cities using solar energy. Sustain Energy Technol Assessments 46. https://doi.org/10.1016/j.seta.2021.101281
10. Sathyakala MD, Suganya MK (2020) Smart bin for waste management and pollution control. 2(4):22–26
11. Bhatt MC, Sharma D, Chauhan A (2019) Smart dustbin for efficient waste management (7):967–969
12. Sreejith S, Ramya R, Roja R, Sanjay Kumar A (2019) Smart bin for waste management system. In: 2019 5th International conference advanced computer communication system ICACCS 2019, pp 1079–1082. https://doi.org/10.1109/ICACCS.2019.8728531
13. Arooj A, Farooq MS, Akram A, Iqbal R, Sharma A, Dhiman G (2021) Big data processing and analysis in internet of vehicles: architecture, taxonomy, and open research challenges (5)
14. Zhou Y et al (2021) Urban rain flood ecosystem design planning and feasibility study for the enrichment of smart cities. Sustain 13(9):1–15. https://doi.org/10.3390/su13095205
15. Atayero AA, Popoola SI, Williams R, Badejo JA, Misra S (Dec 2019) Smart city waste management system using internet of things and cloud computing. In: International conference on intelligent systems design and applications. Springer, Cham, pp 601–611
16. Souza LS, Misra S, Soares MS (July 2020) SmartCitySysML: a SysML profile for smart cities applications. In: International conference on computational science and its applications. Springer, Cham, pp 383–397
17. Olowu M, Yinka-Banjo C, Misra S, Oluranti J, Ahuja R (Dec 2019) Internet of things: demystifying smart cities and communities. In: International conference on advances in computational intelligence and informatics. Springer, Singapore, pp 363–371
18. Firat Y, Uğurlu T (2019) Automatic garage door system with arduino for defined licence plates of cars. 2018 International conference artificial intelligence data processing IDAP 2018. https://doi.org/10.1109/IDAP.2018.8620835

Futuristic Computing Technologies

Randomly Hiding Secret Data Using I-Blocks and E-Blocks for Image Steganography

Anilkumar Patel and Daxa Vekariya

Abstract Various methods to protect the data over transmission channel like cryptography and steganography are analyzed. Steganography is covered writing. One can hide secret data in image, audio or video called cover. To hide data in covers having different formats, with high embedding payload and still maintaining visual imperceptibility, is a challenging task. Security, robustness and payload of secret data play a major role in deciding the approach for hiding secret data. In proposed approach, the cover image is partitioned into blocks each of 4 × 4 pixels. The blocks are divided into index block called I-blocks and embed blocks called E-blocks. I-blocks are used to hide the index of selected pixels of E-blocks. Embed blocks are the blocks in which to hide the secret information. The approach works fine for different image formats as well as for hiding large text over color cover image with high imperceptions.

Keywords Data hiding · E-blocks · I-blocks · Steganography

1 Introduction

In today's world, people communicate over the Internet and share private information. In order to block data from intruders and hackers, this secret information should be protected through a secure technique [1]. The secret data can be hacked for the purpose of copyright violation, for tampering it or can be illegally accessed without the knowledge of owner [2]. Due to these reasons, there is a need of hiding secret data inside different types of digital data such that owner can prove copyright ownership, identify attempts to tamper with sensitive data and to embed annotations.

A. Patel (✉) · D. Vekariya
Department of Computer Science and Engineering, PIET, Parul University, Vadodara, Gujarat, India
e-mail: anilpatel11@gmail.com

D. Vekariya
e-mail: daxa.vekariya18436@paruluniversity.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_27

Steganography is covered writing [3]. It is a process that involves hiding important information (message) inside other carrier (cover) data to protect the message from unauthorized users. The message and the cover data can be of any format such as text, audio, image and video. As per [4], the combined method of least significant bit (LSB) and dynamic management position was proposed for providing hidden images as the secret message. A good steganography system should have high embedding payload and high embedding efficiency. First, the embedding payload is defined as the amount of secret information that is going to be embedded inside the cover data. A low modification rate and good quality of the cover data lead to a high embedding efficiency [5]. Ref. [6] proposed the randomness of data hiding in the cover image. According to [7], the major challenges in image steganography can be noiselessness in higher level, security for hidden data and high payload capacity. As per [8], four (Jsteg, F3, F4 and F5) JPEG steganography techniques available for image steganography. The most important parameters are embedding capacity, robustness and un-detectability [9]. In [10], Essam H. Houssein et al. proposed an advanced technique for encrypting data using advanced encryption system (AES) and hiding the data using Haar discrete wavelet transform (HDWT). Horng [11] proposed quotient value differencing (QVD) and least significant bits (LSBs) substitution to hide secret data in an absolute moment block truncation coded (AMBTC) image for increased embedding capacity. This scheme achieves the best embedding capacity in comparison with existing AMBTC-based methods using the same PSNR level [11]. Kaur [12] proposes a hybrid algorithm for robust image steganography. The Hcf coefficients are utilized for maintaining the perceptual quality of the image [12]. Wei Lu [13] proposed secure halftone image steganography based on pixel density transition. It introduces the concept of black pixel density into steganography of halftone images [13]. Ref. [14] proposes deep learning method to achieve image steganography. The training process is time and power consuming and takes about 3 to 4 days of computation on a 4 GB graphics card [14]. Algorithm proposed in [15] decreases the distortion in the cover image.

2 Related Work: A Brief Description of Energy Matrix, Cost Matrix, Random Traversal and Data Embedding Procedure

2.1 Energy Matrix

Consider an image I with a 4 × 4 dimension. An eight-bit image pixel with values in the range of 0 to 255 has been used. Let $I(x, y)$ be the position of a pixel. A generalized formula for an energy function is as per following equations:

$$\text{Xenergy} = I(x-1, y-1) + 2I(x, y-1) + I(x+1, y-1)$$

$$- I(x-1, y+1) - 2I(x, y+1) - I(x+1, y+1) \tag{1}$$

$$\begin{aligned}\text{Yenergy} = {} & I(x-1, y-1) + 2I(x-1, y) + I(x-1, y+1) \\ & - I(x+1, y-1) - 2I(x+1, y) - I(x+1, y+1)\end{aligned} \tag{2}$$

$$E = \sqrt{(\text{Xenergy})^2 + (\text{Yenergy})^2} \tag{3}$$

The energy map E thus obtained for the image [6].

The image I (4 × 4):

90	234	97	135
212	73	145	199
149	193	19	239
140	192	13	33

The energy matrix E:

734.6360	505.7845	566.5933	640.1328
578.1500	276.1340	219.6588	426.3051
651.4799	455.2713	316.5470	503.6983
757.6345	674.0712	543.0322	499.0331

2.2 Cost Matrix

For a given pixel $E(x, y)$, look for its neighboring pixels in the vertical upward direction and find a new value for that particular pixel. Accommodate all such values to form a new matrix which is called the cost matrix C [6]. For a pixel $E(x, y)$, we check for the pixels $E(x - 1, y - 1)$, $E(x - 1, y)$ and $E(x - 1, y + 1)$. Among these three pixels, take the value of the pixel which has minimum energy. Now, add this value to $E(x, y)$ to obtain the cost of the pixel with position (x, y) as $C(x, y)$. This procedure is followed for the entire energy matrix E, and thus, we calculate the cost matrix C [6].

The cost matrix C:

734.4	505.8	566.6	640.1
1083.9	781.9	725.4	992.9
927.6	674.9	536.2	723.3
1212.9	990.6	859.5	815.5

2.3 Random Traversal Procedure

Random traversal procedure is employed to cost matrix C to select the pixels in which secret binary digits (bits) need to be embedded. In each row, one pixel is selected, which totals four per matrix. {90 212 149 140} are the pixels in which data is to be embedded [6].

2.4 Data Embedding Procedure

Simple LSB substitution has been used. Let $Q = 3$ be the number of bits embedded in each pixel. {1 0 0 1 0 0 1 1 0 0 1 1} be a stream of 12 binary digits to be embedded in the cover image. The 3 LSB of each of the four pixels is altered to get the stego image as follows: [6].

92	234	97	135
212	73	145	199
150	193	19	239
139	192	13	33

The advantage of this approach is it allows random selection of the pixel for embedding. On the other hand, the algorithm works fine for embedding text in mentioned gray images without loss of perception but does not give desired results for embedding text, gray image and color image in some color cover images. In fact, the proposed strategy works fine with the embedding algorithm, but the same strategy does not work for extracting algorithm for some color images. Need to cover color cover images and overcome the limitation of existing algorithm motivate us to develop the proposed approach.

3 Proposed Approach

The existing algorithm has been tested on five different standard images: cameraman, football, girl, coins and board. All these images are gray images.

In proposed approach, the cover image is partitioned into blocks each of 4 × 4 pixels. The blocks are divided into index block called I-blocks and embed blocks called E-blocks. I-blocks are used to hide the index of selected pixels of E-blocks. Embed blocks are the blocks in which to hide. The selection of pixel for embedding secret data is based on energy matrix, cost matrix and random traversal procedure of the related work.

We proposed the following embedding and extracting algorithm for hiding secret message (text/gray image/color image) in color cover image.

3.1 Embedding Algorithm

(1) Consider M x N color image as cover image. (We can have 512 × 512, 256 × 256, 128 × 128, 64 × 64).

(2) Partition cover image into M/4 × N/4 total blocks. Each block being 4 × 4 dimensional matrixes made of 16 pixels.

(3) Divide cover image blocks into I-blocks and E-blocks groups.

We consider first 25% blocks as I-blocks and remaining 75% blocks as E-blocks. The pixels in which to embed into E-block are found using steps 5 to 8, and its index is stored in I-blocks.

(4) Convert secret message (text/image) into stream of binary bits. Obtain the size of secret message (text, image) to hide. Append secret message (text, image) pixel values to its size. Convert it to binary. Do padding, if required. (Four pixels in each block are selected; hence, 12 bits are embedded in each block). Pad reference: 12 bits.

(5) Determine the energy matrix E for each E-block of the color image.

(6) Calculate the cost matrix C from energy matrix E.

(7) On cost matrix: Route the path using random traversal procedure. Four pixels are selected for each 4 × 4 E-block. Index and pixel values are preserved.

(8) Using simple LSB substitution method, secret data is embedded into four pixels in each of 4 × 4 E-blocks. Use 3-LSB bits for embedding.

(9) Embed the index values of selected pixels into I-blocks. The index values are in terms of row-column pair. Consider that row and columns start from 0. If in, zeroth row-first pixel, first row-second pixel, second row-third pixel and third row-zeroth pixels are selected, then index values obtained are (0, 1), (1, 2), (2, 3) and (3, 0). Considering rows part implicit, the index obtained is (1, 2, 3, 0). The index values for selected pixels 90, 212, 149 and 140 in terms of rows and columns for the above given example are (0, 0), (1, 0), (2, 0) and (3, 0). Using only column part, index is (0, 0, 0, 0). For storing index values, we use two LSB's.

Consider 4 × 4 I-block as follows:

16	15	14	13
12	11	10	9
8	7	6	5
4	3	2	1

$$16 = 00010000 \qquad 00010000 = 16$$
$$15 = 00001111 \qquad 00001100 = 12$$
$$14 = 00001110 \qquad 00001100 = 12$$

$$13 = 00001101 \qquad 00001100 = 12$$

After embedding:

16	12	12	12
12	11	10	9
8	7	6	5
4	3	2	1

3.2 *Extracting Algorithm*

(1) Obtain M × N (64 × 64) stego image.
(2) Partition stego image into M/4 × N/4 total blocks.
(3) Divide into I-blocks and E-blocks.
(4) Extract I-blocks to obtain the index.
(5) Extract E-blocks to obtain stream of binary bits (Figs. 1 and 2).

4 Result and Analysis

The performance of the algorithm is measured in terms of PSNR, SSIM and UIQI. PSNR is peak signal-to-noise ratio and it measures in decibel. PSNR gives measure of quality of the stego image. SSIM is structure similarity index and it measures image degradation as a perceived change in structural information. UIQI is universal image quality index and it measures distortion that has occurred in cover object due to embedding process (Tables 1, 2, 3 and 4).

Table 5 gives performance metrics for text in image steganography and Table 6 gives performance metrics for secret color image in image steganography for number of bits embedded = 3. The results show that PSNR is in the range of 40 to 60 db for hiding secret text and in the range of 40 to 50 db for hiding secret image. We know that PSNR depends on the pixels modified in the cover image, hence, if less bytes are embedded, we get higher PSNR. Again, if the cover image has more pixels with last three bits similar to the bits to be embedded, than higher PSNR is obtained. Also, for the same cover image, we get different PSNR for different secret message/image. This is given in Table 5. The concept of sending the size of the secret message makes it easy to extract number of bits embedded. The proposed approach was tested on various size of cover image. The secret texts containing alphanumeric and special characters were tested. It was also tested for different size of secret text files. It supports secret text up to 999,999 characters (999.999 KB). By extending the number of blocks for hiding secret message size, embedding size can be increased to more than 1 MB. The results show that the performance of the algorithm is consistent for different

Fig. 1 Block diagram of embedding algorithm (Secret color image over color image). *Source* Internet

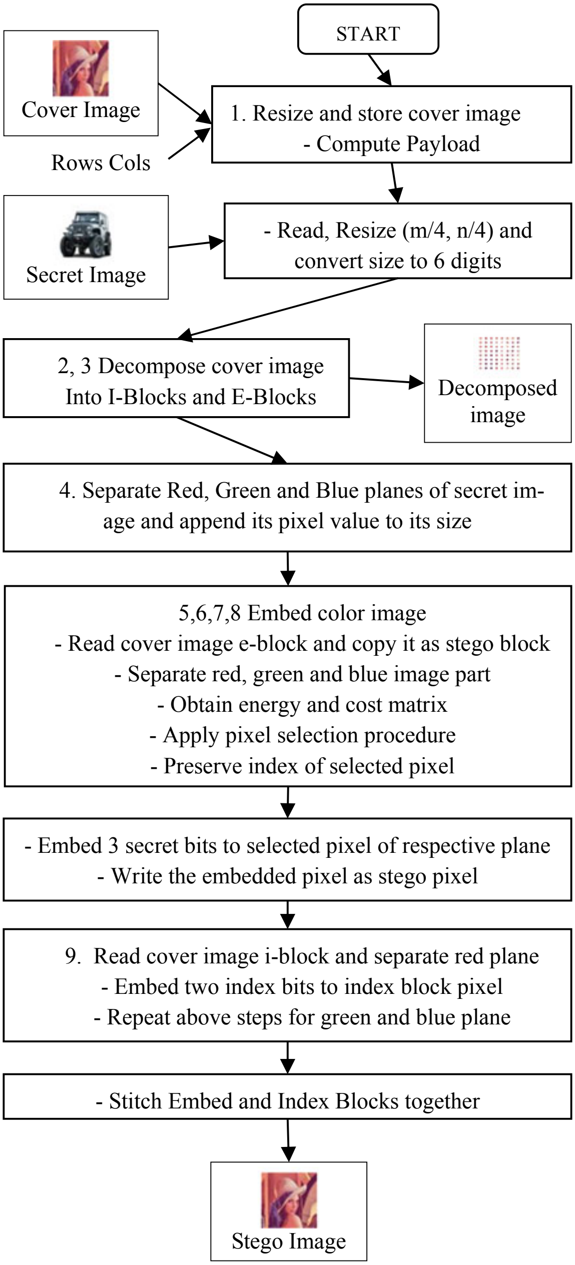

START
Cover Image
Rows Cols
1. Resize and store cover image
- Compute Payload
Secret Image
- Read, Resize (m/4, n/4) and convert size to 6 digits
2, 3 Decompose cover image Into I-Blocks and E-Blocks
Decomposed image
4. Separate Red, Green and Blue planes of secret image and append its pixel value to its size
5,6,7,8 Embed color image
- Read cover image e-block and copy it as stego block
- Separate red, green and blue image part
- Obtain energy and cost matrix
- Apply pixel selection procedure
- Preserve index of selected pixel
- Embed 3 secret bits to selected pixel of respective plane
- Write the embedded pixel as stego pixel
9. Read cover image i-block and separate red plane
- Embed two index bits to index block pixel
- Repeat above steps for green and blue plane
- Stitch Embed and Index Blocks together
Stego Image

Fig. 2 Block diagram of extracting algorithm. *Source* Internet

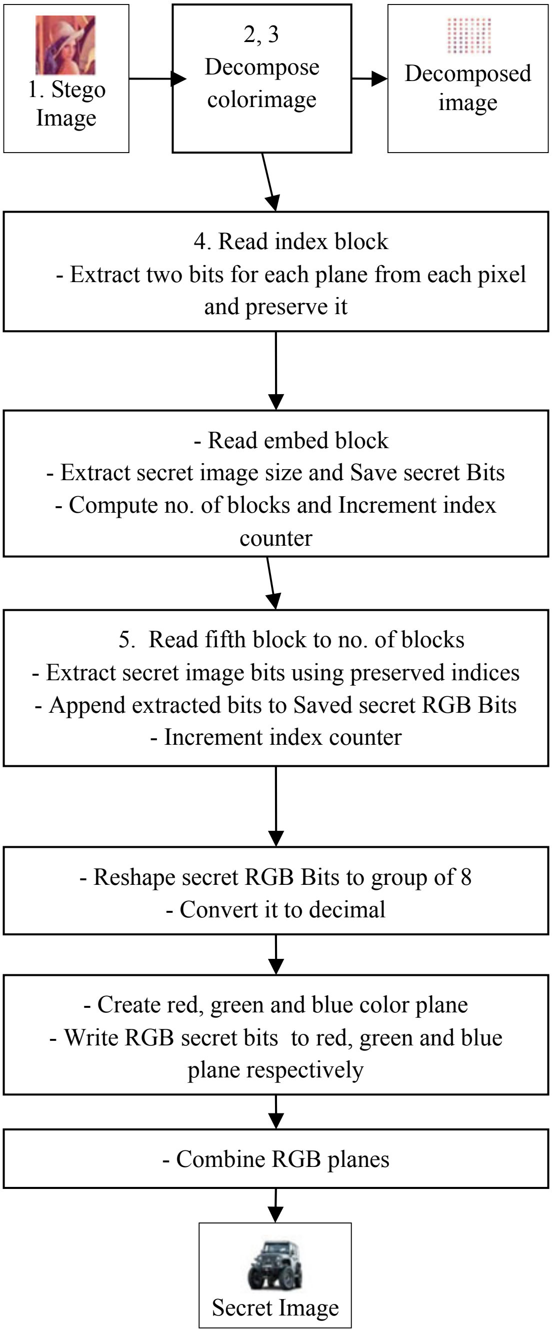

1. Stego Image
2, 3 Decompose colorimage
Decomposed image
4. Read index block
- Extract two bits for each plane from each pixel and preserve it
- Read embed block
- Extract secret image size and Save secret Bits
- Compute no. of blocks and Increment index counter
5. Read fifth block to no. of blocks
- Extract secret image bits using preserved indices
- Append extracted bits to Saved secret RGB Bits
- Increment index counter
- Reshape secret RGB Bits to group of 8
- Convert it to decimal
- Create red, green and blue color plane
- Write RGB secret bits to red, green and blue plane respectively
- Combine RGB planes
Secret Image

Table 1 Sample images used as cover-secret image

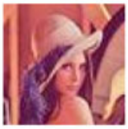
b1.bmp	b2.bmp	gr1.jpg	gr2.jpg
j4.jpg	j5.jpg	 j8.jpg	j11.jpg
j19.jpg	j15.jpg	p3.png	p5.png
p9.png	w5.jpg	t3.tif	t5.tif
w2.jpg	w4.jpg	w6.jpg	w8.jpg

Table 2 Text files as secret text

File name	Content
1. txt (80 bytes)	Randomly hiding secret data using E-blocks and I-blocks for image steganography
2. txt (262 bytes)	This file is created to hide text……
3. txt (320 bytes)	……. Please change the size of the cover image
4. txt (2515 bytes)	Sachin Tendulkar was born April 24, 1973 ………… period of one year and 109 days only
5. txt (10414 bytes)	Writings: …. to thank them for the trust they have in me. [77]

Table 3 Sample stego images

Cover Image	Secret Message	Stego Image
64x64b1.bmp	1.txt (000080 Bytes)	64x64b1.bmpchars000080.png
128x128j4.jpeg	32x32j11.jpg	128x128j4.jpeg32x32j11.jpg.png
256x258w8.jpg	64x64t5.tif	256x256w8.jpg64x64t5.tif.png

Table 4 Nomenclatures for Tables 5, 6 and 7

CIF	Cover image format
CIS	Cover image size in bytes
CIC	Cover image colors
SMS	Secret message size in bytes
PSNR	Peak signal-to-noise ratio in db
SSIM	Structure similarity index
UIQI	Universal image quality index
SMPL	Secret message payload in bytes
MCPL	Maximum cover payload in bytes

image types, e.g., jpg, png, tiff, bmp (gray). Experiments have been done using noise models like Gaussian, Poisson, Salt and Pepper and Speckle for stego image. In case if stego image contains secret image and if salt-pepper noise is added over stego image, then also it gives perceptible results. Experiments show that if we change brightness and contrast of the stego image, the secret image can be extracted using trial and error but it is vulnerable against the compression attack.

5 Conclusion

A new, efficient and secure steganographic method has been proposed. I-block and E-block partition make the proposed method comparatively secure. Randomly selecting only four pixels from a block improves quality of stego image. Fewer bits lead less capacity and more bits degrade the image. 3 bits are used. It supports secret text up to

Table 5 Performance metrics for text in image steganography

S. N.	Stego image	CIF	CIS		CIC	SMS	PSNR	SSIM	UIQI	SMPL	MCPL
1	64 × 64b1.bmp	bmp	64	64	1	80	49.18124	0.99947	0.99962	80	282
2	64 × 64b1.bmp	bmp	64	64	1	262	44.15183	0.99849	0.99891	262	282
3	64 × 64gr3.jpg	jpg	64	64	3	262	49.05132	0.99774	0.9999	262	282
4	64 × 64j2.jpg	jpg	64	64	3	80	54.05909	0.99954	0.99998	80	282
5	64 × 64p5.png	png	64	64	3	262	48.54859	0.99741	0.99993	262	282
6	64 × 64t4.tif	Tif	64	64	1	262	44.20217	0.99752	0.99973	262	282
7	64 × 64w1.jpg	jpg	64	64	3	80	53.80509	0.99973	0.99998	80	282
8	128 × 128p9.png	png	128	128	1	320	47.40774	0.99895	0.98848	320	1146
9	256 × 256t3.tif	Tif	256	256	1	2515	46.7885	0.9966	0.99988	2515	4602
10	512 × 512w6.jpg	jpg	512	512	3	10,414	51.39399	0.99795	0.99739	10,414	18,246
					Average		**48.85895**	**0.99834**	**0.99838**		

MCPL = 3 * MCPL for color cover images. It supports bmp gray cover images. It supports jpg, png and tif color/gray cover images. Secret message is embedded only in the first plane of RGB cover

Table 6 Performance metrics for secret color image in image steganography

S. N.	Stego image	CIS		SCIS		PSNR	SSIM	UIQI	SMPL	MCPL
1	128 × 128j4.jpeg	128	128	32	32	44.10857	0.99428	0.99685	3072	3438
2	128 × 128j9.jpg	128	128	32	32	44.12821	0.98184	0.99964	3072	3438
3	128 × 128j12	128	128	32	32	44.53442	0.98482	0.99798	3072	3438
4	128 × 128j1	128	128	32	32	44.43935	0.98883	0.99290	3072	3438
5	256 × 256j5.jpg	256	256	64	64	44.78700	0.98717	0.99983	12,288	13,806
6	256 × 256p4.png	256	256	64	64	43.10320	0.94848	0.94624	12,288	13,806
7	256 × 256p5.png	256	256	64	64	43.33720	0.97423	0.99472	12,288	13,806
8	256 × 256j1.jpg	256	256	64	64	43.91488	0.98386	0.99215	12,288	13,806
9	512 × 512j7.jpg	512	512	128	128	44.60331	0.99275	0.99779	49,152	55,278
10	512 × 512w4.jpg	512	512	128	128	44.87400	0.99659	0.99989	49,152	55,278
11	64 × 64j2.jpg	64	64	16	16	44.27195	0.99630	0.99972	768	846
12	64 × 64j8.jpg	64	64	16	16	44.56444	0.99586	0.99953	768	846
13	64 × 64t2.tiff	64	64	16	16	44.35560	0.99554	0.99960	768	846
			Average			**44.23247**	**0.98620**	**0.99360**		

SCIS = Secret color image size in bytes. It supports jpg, png, tif as cover images. It supports jpg, png, and tiff as secret images

Table 7 Performance metrics for secret gray image in image steganography

S. N.	Stego image	CIS		CIC	SIS		PSNR	SSIM	UIQI	SMPL	MCPL
1	256 × 256w8.jpg	256	256	3	64	64	49.20384	0.99302	0.99975	4096	13,806
2	256 × 256t1.tif	256	256	1	64	64	44.69605	0.9816	0.99798	4096	13,806
3	256 × 256t1.tif	256	256	1	64	64	44.59757	0.98123	0.99789	4096	13,806
4	128 × 128p1.png	128	128	1	32	32	44.69584	0.99422	0.99968	1024	3438
5	128 × 128p3.png	128	128	3	32	32	48.70921	0.99765	0.99769	1024	3438
6	128 × 128p5.png	128	128	3	32	32	49.12634	0.99574	0.99993	1024	3438
7	128 × 128p8.png	128	128	1	32	32	44.55466	0.9859	0.99964	1024	3438
8	256 × 256t2.tif	256	256	3	64	64	49.08864	0.99434	0.99997	4096	13,806
9	128 × 128j1.jpg	128	128	3	32	32	49.44428	0.99648	0.99808	1024	3438
10	128 × 128gr2.jpg	128	128	3	32	32	49.48862	0.99923	0.99804	1024	3438
11	256 × 256b1.bmp	256	256	1	64	64	44.52304	0.99141	0.99852	4096	13,806
12	256 × 256b2.bmp	256	256	1	64	64	44.63166	0.99728	0.99983	4096	13,806
13	128 × 128b1.bmp	128	128	1	32	32	43.96727	0.99479	0.99871	1024	3438
14	128 × 128b2.bmp	128	128	1	32	32	43.58779	0.99475	0.99982	1024	3438
15	128 × 128b3.bmp	128	128	1	32	32	44.55745	0.99073	0.9998	1024	3438
16	128 × 128j6.jpg	128	128	3	32	32	49.37019	0.99534	0.99907	1024	3438
17	128 × 128j16.jpg	128	128	3	32	32	49.07849	0.99597	0.99981	1024	3438
18	128 × 128j14.jpg	128	128	3	32	32	49.17944	0.99536	0.99989	1024	3438
19	128 × 128t5.tif	128	128	1	32	32	44.57425	0.9879	0.99994	1024	3438
20	128 × 128t3.tif	128	128	1	32	32	44.74151	0.99584	0.99978	1024	3438

(continued)

Table 7 (continued)

S. N.	Stego image	CIS		CIC	SIS		PSNR	SSIM	UIQI	SMPL	MCPL
21	128 × 128t4.tif	128	128	1	32	32	44.12814	0.99746	0.99971	1024	3438
22	512 × 512j15.jpg	512	512	3	128	128	49.24874	0.99814	0.99944	16,384	55,278
23	512 × 512j17.jpg	512	512	3	128	128	49.31263	0.99748	0.99994	16,384	55,278
24	64 × 64w5.jpg	64	64	3	16	16	44.10200	0.99462	0.99967	256	846
25	64 × 64w4.jpg	64	64	3	16	16	49.10804	0.99806	0.99996	256	846
26	128 × 128t1.tif	128	128	1	32	32	44.47934	0.9839	0.99741	1024	3438
27	256 × 256t1.tif	256	256	1	64	64	44.34167	0.98029	0.99775	4096	13,806
28	512 × 512t1.tif	512	512	1	128	128	44.57881	0.99298	0.99768	16,384	55,278
29	128 × 128p8.png	128	128	1	32	32	44.55466	0.9859	0.99964	1024	3438
30	256 × 256p8.png	256	256	1	64	64	44.53851	0.97936	0.9996	4096	13,806
31	512 × 512p8.png	512	512	1	128	128	43.01615	0.98846	0.99948	16,384	55,278
						Average	**46.23310**	**0.99210**	**0.99920**		

SIS = Secret image size in bytes. Secret image is embedded only in the first plane of RGB cover. It supports jpg, png and tif color/gray cover images. It supports bmp gray cover images. It supports jpg, png, tif and bmp gray secret images. Actually, it converts color secret to gray secret

999,999 characters. One can extend beyond it. The average PSNR is 48.85895, SSIM is 0.99834 and UIQ is 0.99838. This shows stego image is still highly imperceptible. Image format supported are: jpg, png, tiff and bmp (gray) for image steganography. The existing system uses same concept for hiding and extracting purpose. We now that after secret bits are embedded, embedded pixel value gets altered and if same extraction process is used, for certain images, it may select incorrect pixel. In the proposed system, extraction process only uses index array from I-blocks leaving least chance of incorrect pixel selection. The proposed approach was tested on different combinations in terms of size of cover image, size of secret text, type of cover and secret image format.

6 Future Work

Although system is secure and works well with diversified images, it is required to make it more robust against the attacks like compression. Proposed approach supports single frame gray gif format. It can be extended to multi-frame color gif format.

References

1. Kaur G (2017) A review: network security based on cryptography and steganography techniques. Int J Adv Res Comput Sci 8(4)
2. Vyas A, Dudul S (2015) An overview of image steganographic techniques. Int J Adv Res Comput Sci 6(5)
3. Steganography. https://en.wikipedia.org/wiki/Steganography
4. Isnanto RR, Septiana R, Hastawan AF (Nov 2018) Robustness of steganography image method using dynamic management position of least significant bit (LSB). In: 2018 International seminar on research of information technology and intelligent systems (ISRITI). IEEE, pp 131–135
5. Mstafa RJ, Elleithy KM (2016) A video steganography algorithm based on Kanade-Lucas-Tomasi tracking algorithm and error correcting codes. Multimedia Tools Appl 75(17):10311–10333
6. Khandelwal P, Bisht N, Thanikaiselvan V (Dec 2015) Randomly hiding secret data using dynamic programming for image steganography. In: 2015 International conference on computing and network communications (CoCoNet). IEEE, pp 777–783
7. Kadhim IJ, Premaratne P, Vial PJ, Halloran B (2019) Comprehensive survey of image steganography: techniques, evaluations, and trends in future research. Neurocomputing 335:299–326
8. Kim JT, Kim S, Kim K (Oct 2019) A study on improved JPEG steganography algorithm to prevent steganalysis. In: 2019 international conference on information and communication technology convergence (ICTC). IEEE, pp 960–963
9. Watni D, Chawla S (Oct 2019) A comparative evaluation of jpeg steganography. In: 2019 5th international conference on signal processing, computing and control (ISPCC). IEEE, pp 36–40
10. Houssein EH, Ali MA, Hassanien AE (Sep 2016) An image steganography algorithm using haar discrete wavelet transform with advanced encryption system. In: 2016 federated conference on computer science and information systems (FedCSIS). IEEE, pp 641–644

11. Horng JH, Chang CC, Li GL (2020) Steganography using quotient value differencing and LSB substitution for AMBTC compressed images. IEEE Access 8:129347–129358
12. Kaur R, Singh B (2021) A hybrid algorithm for robust image steganography. Multidimension Syst Signal Process 32(1):1–23
13. Lu W, Xue Y, Yeung Y, Liu H, Huang J, Shi Y (2019) Secure halftone image steganography based on pixel density transition. IEEE Trans Dependable Secure Comput
14. Kumar V, Laddha S, Aniket ND (2020) Steganography techniques using convolutional neural networks. J Homepage 7(3):66–73 http://iieta.org/journals/rces
15. Radeaf HS, Mahmmod BM, Abdulhussain SH, Al-Jumaeily D (April 2019) A steganography based on orthogonal moments. In: Proceedings of the international conference on information and communication technology, pp 147–153

On the Efficacy of Deep Image Denoising for Computer Vision Applications

Manan Shah and Pankaj Kumar

Abstract Image denoising is a process of inverse reconstruction where the original image is reconstructed from its noisy observations. Several deep learning models have been developed for image denoising. Usually, the performance of image denoising is measured by metrics like structural similarity index (SSIM) and peak signal-to-noise ratio (PSNR), however in this paper, we take a more pragmatic approach. We design and conduct experiments to evaluate the performance of deep image denoising methods in terms of improving the performance of some popular computer vision (CV) algorithms after image denoising. In this paper, we have comparatively analyzed: fast and flexible denoising (FFDNet) convolution neural network (CNN), feed forward denoising CNN (DnCNN), and deep image prior (DIP)-based image denoising. CV algorithms experimented with are face detection, face recognition, and object detection. Standard and augmented datasets were used in our experiments. Various types and amounts of noise were added to raw images from standard datasets (BSDS500, LFW, FDDB, and WGSID). We may conclude from our findings that image denoising does not improve the performance of CV algorithms when applied to raw images of datasets. But image denoising is very effective in improving the performance of the CV methods when denoising is applied to noise corrupted images of the datasets. In our experiments, we found results where the improvements were up to 11.70% in terms of accuracy for the face detection experiment.

Keywords Image denoising · Computer vision · Face recognition · Face detection · Object detection · Convolution neural network · Deep learning

M. Shah (✉)
Dhirubhai Ambani Institute of Information Communication Technology, Gandhinagar, India
e-mail: 201911028@daiict.ac.in

P. Kumar
University Petroleum and Energy Studies, Bidholi, India
e-mail: pankaj.k@ddn.upes.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_28

1 Introduction

Removing noise from an image captured by visual sensors is one of the most fundamental challenges in image processing and computer vision (CV) tasks. The key objective here is to approximate the original image based on the noisy observation. Images are inevitably polluted by noise during capture, compression, and transmission due to the impact of the environment, transmission channel, and other factors, resulting in distortion and loss of image data [1]. Some operations, such as image analysis, tracking, and video processing, are affected by noise. Many practical applications of CV, such as face detection [2], face recognition [3], object detection [4], and target tracking [5], are affected by noise for several reasons.

As we said, the quality of an image is influenced by various sensors and artifacts noise as it is being captured. As a result, dealing with noise is an important part of image processing and computer vision. With the increased usage of deep learning, image denoising algorithms can achieve promising results. Due to the requirement of good image quality and its usability in certain practical applications, image denoising is becoming a fascinating area of research. Image denoising is an inverse image reconstruction problem, in which we try to recover original image x from noisy image y, where y is mathematically defined as $y = x + \sigma$, where one assumption is that σ is additive white Gaussian noise (AWGN) with standard deviation σ [6] (Fig. 1).

FFDNet [8], DnCNN [9], and DIP [10] are three deep learning-based image denoising methods that we evaluated. The peak signal-to-noise ratio (PSNR) [11], the structural similarity index (SSIM) [11], and the blind/referenceless image spatial quality evaluator (BRISQUE) [12] were used to compare the output results of image denoising techniques. Face recognition methods such as eigenface [2] and face detection algorithms such as Viola–Jones [3] are used. Face recognition and face detection results are compared and shown using accuracy. The yolov5x [13] method is utilized to detect objects, and the result is given using mean average precision (mAP) [14] (Fig. 2).

For image denoising, FFDNet [8] is the first technique to incorporate rectified linear unit (ReLU) [16] and batch normalization [17]. It can achieve a reasonable balance between inference speed and denoising performance, and it can efficiently handle a wide range of noise levels [0–75] (σ) inside a single network. By giving a

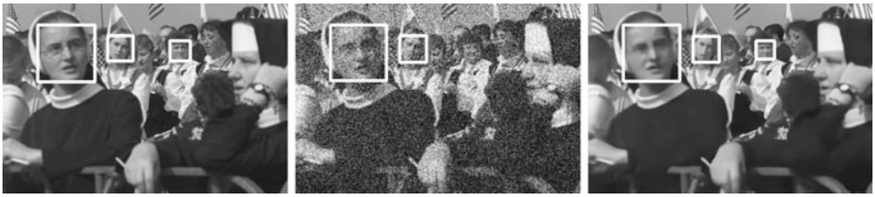
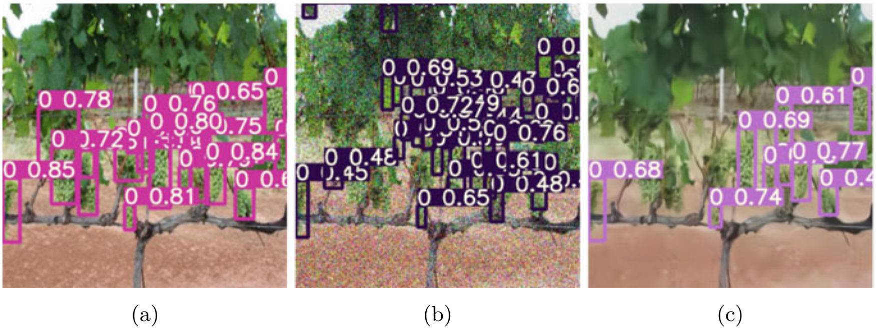


Fig. 1 **a** Face detection on a raw image from FDDB [7] dataset. **b** Face detection results are degraded according to image noise. Take note of one missed detection. **c** Face detection improved when noisy image is denoised by FFDNet [8]



(a) (b) (c)

Fig. 2 **a** Object detection on a raw image from WGSID [15] dataset. **b** As the number of false detections increases, the object detection result decreases. **c** Reduction in false detection when noisy image is denoised by FFDNet [8]

non-uniform noise level map, FFDNet [8] can reduce spatially variable noise. It is faster than block-matching and 3D filtering (BM3D) [18], which is the state-of-the-art approach on CPU and GPU, without sacrificing denoising performance.

DnCNN [9] is the first image denoising method to utilize skip connection [19]. This algorithm works on Gaussian denoising with the unknown noise level, i.e., σ not known in advance. In addition to image denoising, the DnCNN approach is effective for single-image super-resolution and JPEG image deblocking.

DIP [10] is a new method that bridges the gap between learning-based deep convolutional methods and learning-free methods. Randomly initialized neural networks can be utilized as a handcrafted prior in the DIP algorithm, resulting in an outstanding performance in a variety of image processing tasks. The image denoising task is performed with DIP as part of this research.

Face recognition system has many practical applications like attendance systems, social media photo tagging, authentication in companies, phone unlock, etc. Eigenface [2] was the first-ever breakthrough in the face recognition system. It is a simple, fast, and efficient method and provides good results on small datasets too.

Face detection is the first step in face recognition and it is useful for detecting a face in the images. It is a subpart of object detection. It has several practical applications like biometrics, security, law enforcement, etc. Viola–Jones [3] algorithm works in real time to detect faces and provides high accuracy.

Object detection is a technique for locating items in a video or picture. It is also one of the most active disciplines and has numerous practical applications. We are using YOLOv5x [13] to detect grapes here. It is the most recent and fastest version of you only look once (YOLO) [4]. In the agricultural area, grape detection [15] has certain useful applications.

Due to the practical importance of CV, reducing noise in the image while preserving important features like edge, structure, and corners become an important challenge. Some other challenging tasks contain smoothing the flat area, protecting the edge without making it blurry, preserving texture, and not generating any new artifacts [1]. The main contribution of our study is to find out how CV applications are affected by noise and how deep image denoising algorithms help to improve it.

The objective of this research is to improve the performance of computer vision applications. In everyday life, face recognition [2], face detection [3], and object detection [4] are used. These applications suffer from noise over time as a result of hardware or sensor issues. We are attempting to demonstrate that the performance of these applications can be improved when image quality deteriorates due to noise in the image. When image quality is adequate and no further improvement is possible, this research is limited.

From various user perspectives, research is beneficial in the specific application that deals with images. One example is the noise or blur effect that occurs when fruits are captured by automation via robots or moving vehicles [20].

To our knowledge, this is the first study that ever was looking at the impact of deep picture denoising on CV applications. The remainder of the paper is laid out as follows: The mechanism of our approach is explained in Sect. 2. In parts 3 and 4, you will learn about performance metrics and datasets. Finally, we go over the experimental setup and data analysis in Sects. 5 and 6. In Sect. 7, we discuss and reach some conclusions.

2 Methodology

In Fig. 3a, b, we show the processing pipelines of the two methodologies used in our approach. In the first approach, we feed our deep image denoising algorithms raw images from standard datasets including LFW [21], FDDB [7], and WGSID [15]. CV techniques such as eigenface [2], Viola–Jones [3], and yolov5x [13] use the denoising algorithm's output. We will get results in terms of a specific vision algorithm's standard metrics. We are comparing the outcome to the CV algorithm's original output (without denoised). The fundamental motivation for using this technique is to see how image denoising affects the behavior of CV applications.

In the second method 3(b), we created an augmented dataset by adding different amount of noise in the images. Image denoising algorithms receive the augmented dataset as input. CV algorithms are fed images that have been denoised. The outcome will be defined in terms of the algorithm's standard metric. We compare our result with the augmented dataset (before denoising) on the CV methods. The fundamental motivation for taking that approach is to see if denoised algorithms could significantly improve the performance of CV applications.

The following are the architecture features of deep image denoising algorithms:

FFDNet [8] takes tunable noise level map and image as input. It downscales the $W \times H \times C$ input image into four $W/2 \times H/2 \times 4C$ downsampled sub-images, where $C = 1$ for grayscale images and $C = 3$ for color images. After that, there are several layers followed by convolution+batch normalization [17] + rectified linear unit (ReLU) [16], and convolution is used in the last layer. As an output, we get

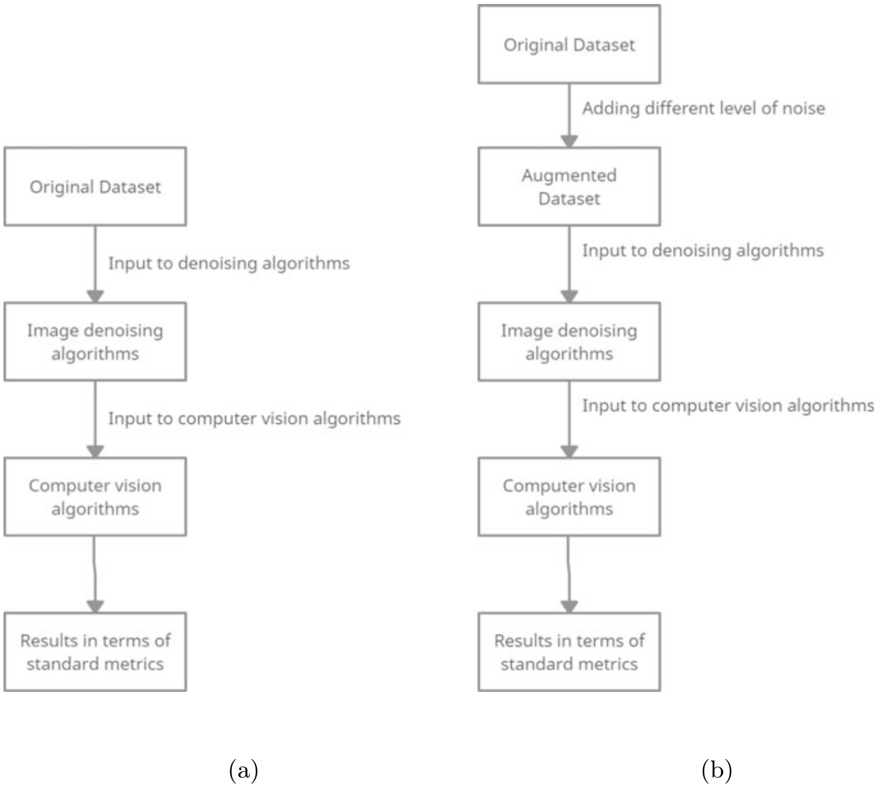

Original Dataset
Input to denoising algorithms
Image denoising algorithms
Input to computer vision algorithms
Computer vision algorithms
Results in terms of standard metrics
(a)
Original Dataset
Adding different level of noise
Augmented Dataset
Input to denoising algorithms
Image denoising algorithms
Input to computer vision algorithms
Computer vision algorithms
Results in terms of standard metrics
(b)

Fig. 3 **a** Processing pipeline when Gaussian noise is not added to images. **b** Processing pipeline when Gaussian noise is added to images

denoised sub-images and as a final output, we get the denoised image. Adam [22] algorithm is adopted here for optimization.

In FFDNet [8], 15 layers and 64 channels are used for grayscale images, while 12 layers and 96 channels are used for color images. Here, orthogonal initialization method for convolution filter is adopted which is helpful for detail preservation and removal of visual artifacts.

In particular, FFDNet [8] is defined as $\hat{x} = F(y, M; \theta)$, where M is input noise level map, $\hat{x}$ is recovered image, y is noisy observation and θ is are fixed parameter for noise level [1]. As it works on sub-sampled images, it can achieve good speed in training and testing.

DnCNN [9] method takes noisy observation and tries to learn mapping functions to predict clean images. Residual learning [19] strategy is used here. It uses convolution + ReLU [16] as the first layer with 64 filters of size $3 \times 3 \times\ C$, where $C = 1$ for grayscale image and $C = 3$ for a colors image. After that, there are several layers followed by convolution + batch normalization [17]+ ReLU [16], and convolution is used in the last layer. Stochastic gradient descent (SGD) with momentum [23] algorithm and Adam [22] algorithm is adopted for optimization.

In particular, DnCNN [9] method learns function $\hat{x} = F(y, \theta_\sigma)$, where θ_σ is parameters for fixed variance σ [1], $\hat{x}$ is recovered image, y is noisy observation, and θ is fixed parameter for noise level. Here, batch normalization [17] and residual learning [19] can benefit each other and helps in speeding up and boosting denoising performance.

DIP [10] algorithm uses encoder-decoder (hourglass) architecture for the image denoising task. Here, denoising problem is simulated as an energy minimization problem. Mean square error (MSE) [24] is used as an optimization or energy function.

In DIP [10], training is done directly on the target image. It takes fixed tensor and random weight as input and learns weight through the feed forward network, and updates weight through gradient descent [23] algorithm. At last, fixed tensor and learned weight are given in feed-forward network to get the final output which is a denoised image.

There are 5 layers used in upsampling (decoder) and downsampling (encoder). No of filters in each layers are 8, 16, 32,64, and 128. At last, two layers skip connection [19] is used. For upsampling bilinear method is used and for downsampling, max-pooling is used.

In particular, DIP [4] method learns the θ parameter with encoder-decoder architecture and updates the θ parameter at every iteration with the use of gradient descent [23] and mean square error [24] as an energy function which is the main contribution of the paper.

Architecture details of Eigenface [2] are as per following.

Eigenface [2] method first represents the image of size $N \times M$ as vector G of size $(N \times M) \times 1$. Then, it finds the mean of the image set and removes it from the individual image. Subtracting the mean from the original image helps us in the removal of common features between images. Then, we apply principal components analysis (PCA) [25] to find eigenvectors of images. These eigenvectors have the same size as original images, so they can be represented as an image and that is why this method is called eigenface [2].

In PCA [25] first, few components contain 90–95% of variance, and that number of components are only needed further for classification. These first eigenvectors work as training data and we can classify by calculating the distance between the target image and eigenvectors.

Architecture details of Viola–Jones [3] algorithm are as per following:

There are four stages in the **Viola–Jones** [3] algorithm. The first step is to calculate a Haar-like feature, which is beneficial for identifying common human characteristics. The eye area is darker than the upper cheeks, and the nasal bridge is darker than the eyes. The integral image assesses rectangular features in constant time in the second step. The AdaBoost classifier is used in the third stage to train and learn the optimal features. The cascade classifier is employed in the fourth phase.

The following is the YOLOv5x algorithm's architecture details:

Backbone, neck, and head are the three primary components of **YOLOv5x** [13]. CSPDarkNet-53 serves as the backbone, and it has been pre-trained using ImagNet

[26] data. It makes advantage of full to maximize gradient flow over the network while keeping computing costs low. Spatial pyramid pooling (SPP) and path aggregation network (PAN) make up the neck component. It aids in expanding the receptive area while maintaining a high network speed. The loss function is binary cross-entropy, and the leaky ReLU and sigmoid activation are utilized.

3 Metrics

The following are the image quality metrics:

- The computation of the ratio between the original image and the noisy image is referred to as **PSNR**. Equation (1) [11] calculates the PSNR between two images a, b. The value of x for an 8-bit input image is 255. MSE stands for mean squared error between matching pixels in two images.

$$\text{PSNR}(a, b) = \frac{10 \log x^2}{\text{MSE}(a, b)} \tag{1}$$

  The greater the PSNR score, the better the image quality.

- **SSIM** The sliding window of size 8×8 is used to calculate SSIM Eq. (2), [11] between two images a, b. The term μ_a and μ_b refer to vector-based 8×8 pixel images.
  $\sigma_a, \sigma_b, and\ \sigma_{ab}$ are co-variances of a and b, respectively.
  $c_1 = (k_1 L)^2$ and $c_2 = (k_2 L)^2$ where L = 7 for 8bit image and k_1 = 0.01 and k_2 = 0.03.

$$\text{SSIM}(a, b) = \frac{(2\mu_a \mu_b + c_1)(2\sigma_{ab} + c_2)}{(\mu_a^2 + \mu_b^2 + c_1)(\sigma_a^2 + \sigma_b^2 + c_2)} \tag{2}$$

  The greater the SSIM, the more comparable the images are, and therefore, the denoising process is improved.

- The **BRISQUE** metric can only be computed with a single picture pixel value. It uses the spatial natural scene statistics (NSS) [12] model of spatially normalized brightness coefficients in the spatial domain. For these coefficients, it also uses a pairwise product model. The following steps are used to compute BRISQUE.
- The mean subtracted contrast normalized (MSCN) coefficient is determined in the first stage.
- Then, by fitting coefficients to a generalized Gaussian distribution, features can be discovered.

- The model is then trained using support vector regression (SVR) utilizing downsampled features and features discovered in the second stage.
- The BRISQUE score is calculated using that trained model. A greater BRISQUE value indicates a worse image quality.

Details of face recognition and face detection metrics are as per following. Here, **TP**: True Positive **FP**: False Positive **TN**: True Negative **FP**: False Positive.

- **Accuracy** is part of prediction which model predicted accurately. It is calculated by using:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{3}$$

The following are the details of the object detection metric:

- Object detection results are reported using **mean average precision (mAP)** [14]. We can demonstrate how accurate bounded boxes are predicted using mAP.
- Precision and recall are calculated in the first phase using intersection over union (IOU). For IOU, a different limit is set.
- The area under the precision-recall curve is used to compute average precision (AP) in the second step.
- The mean value of AP, which is the mean average precision, is computed in the last phase.

4 Dataset

Experiments of deep image denoising algorithms are performed using **Berkeley Segmentation Dataset and Benchmarks 500 (BSDS500)** [27] dataset. It contains a total of 500 images as 200, 200, and 100 images for training, testing, and validation, respectively.

Face recognition experiments are performed using **Labeled Face in Wild (LFW)** [21] dataset. It contains 13000 images from 1680 different individuals. From that, we have taken 1288 images of 7 different individuals. For some of the face recognition experiments, we have also created a manual dataset by adding a different level of noise in the LFW [21] dataset.

For the face detection, we have considered **Face Detection Dataset and Benchmark (FDDB)** [7] dataset to perform experiments. It contains 10 different folders with a total 2845 number of images and 5170 faces. For some of the face detection experiments, we have also created a manual dataset by adding a different level of noise in the FDDB [7] dataset.

Embrapa Wine Grape Instance Segmentation Dataset (WGISD) [15] was used to conduct object detection research. It includes 300 images, separated into 242 for training and validation and 58 for testing.

5 Experiment Setup

The Python programming language was used to code all of the experiments. For some of the experiments, we used a Jupyter notebook from an anaconda distribution package. The majority of our tests used graphics processing unit (GPU), we used Tesla T4 GPU with 12 GB RAM.

Several experiments are being carried out here. The comparison and analysis of the three different deep image denoising techniques are the first experiment. FFDNet [8] and DnCNN [9] are trained on 200 images from the BSDS500 [27] dataset's training folder. Both algorithms were trained on a total of 50000 epochs to train and evaluate the models. The DIP [10] does not require any prior training because it works directly on the target image. DIP [10] runs for 3000 epochs on the target image. We used 200 images from the testing folder of the BSDS500 [27] dataset to test all three algorithms, and the average result is provided in terms of three standard metrics. The outcomes of these studies are given in Tables 1, 2 and 3.

The following experiment is for face recognition using the augmented LFW [21] dataset. We used an augmented dataset as input to the denoising algorithm and performed the eigenface [2] experiment using the output of the denoising method. The outcome of this experiment is compared to the outcome of eigenface [2] on the augmented dataset (without denoising). Because support vector machine (SVM) [28] performed better in the previous studies for face recognition experiments. Therefore, it was our sole consideration for future face recognition experiments. The dataset contains a total of 1288 images and we divided them into 70–30 scales for training and testing. Therefore, the algorithm was trained on 901 images and tested on 387 images. We have considered the first 50 components in principal component analysis (PCA) [25] components which are identical to the second experiment. DIP [10] takes approximately 95–100 h to process 1288 photos on a T4 Tesla GPU with 12 GB of RAM for face recognition experiment on the raw dataset. We did not investigate DIP further for our experiments. Table 4 gives the outcome of this experiment.

The fourth experiment is on face detection using the Viola–Jones [3] algorithm. It is carried out on the FDDB [7] dataset. In this approach, we used a scaling factor of 1.1 and a minimum neighbors count of 3. In addition, we are giving a dataset for denoising the algorithm and redoing the same experiments. As indicated in the facial recognition section, we are not including DIP [10] in our experiment for the same reason. Table 6 gives the results and comparisons of these experiments.

The Viola–Jones [3] technique is used for face detection on the augmented FDDB [3] dataset in the following experiment. The dataset is first preprocessed with image denoising algorithm, and the denoising result is fed into the face detection method. All other parameters are the same as in the previous experiment, and the result is compared to the Viola–Jones [3] method's augmented dataset result (without denoising). Table 5 summarizes the findings of this experiment.

The final experiment is for object detection on the WGISD [15] dataset using yolov5x [13]. Every image has been scaled to 416 × 416. The dataset includes the details of training, testing, and validation images to be evaluated, and specifics are

Table 1 PSNR value comparison for image denoising by FFDNet, DnCNN, and DIP algorithm

	$\sigma = 15$	$\sigma = 25$	$\sigma = 35$
FFDNet	**34.07**	31.39	**29.74**
DnCNN	34.06	**31.40**	29.73
DIP	31.69	29.59	28.65

Bold value indicates highest result with respective of different noise level

Table 2 SSIM value comparison for image denoising by FFDNet, DnCNN, and DIP algorithm

	$\sigma = 15$	$\sigma = 25$	$\sigma = 35$
FFDNet	0.932	0.887	0.847
DnCNN	0.931	0.887	0.848
DIP	**0.999**	**0.999**	**0.999**

Bold value indicates highest result with respective of different noise level

Table 3 BRISQUE value comparison for image denoising by FFDNet, DnCNN, and DIP algorithm

	$\sigma = 15$	$\sigma = 25$	$\sigma = 35$
FFDNet	20.91	23.79	26.34
DnCNN	**15.24**	**15.55**	**15.46**
DIP	22.97	23.06	23.17

Bold value indicates highest result with respective of different noise level

supplied in the object detection dataset section. The model has been trained for 200 epochs. The same dataset was fed into three denoising algorithms, and the experiment was repeated using the results of these methods. The outcome is given in Table 7.

6 Result and Analysis

PSNR [11] is considered as a standard metric for the image denoising benchmark in general. In our experiment, FFDNet [8] and DnCNN [9] both perform better in terms of PSNR [11], with a difference of less than 0.1. FFDNet [8] produces the greatest results for $\sigma = 15$ and 35, while DnCNN [9] produces the best results for $\sigma = 25$. In terms of SSIM [11], DIP [10] findings surpass the other two techniques. Table 3 gives that DnCNN [9] is the clear winner in terms of BRISQUE [9].

The findings of our experiment face detection LFW dataset are given in Table 4. It clearly shows that when noisy images are denoised using image denoising methods, face recognition results improve. In this case, DnCNN [9] outperforms FFDNet [8] on two significant noise levels. We ran further experiments of face detection on the augmented dataset of FDDB, and it is clear that the results are considerably better. As given in Table 5, FFDNet [1] outperforms DnCNN [2] for all three sigma noise levels.

Table 4 Result of Eigenface on output of image denoising algorithm on augmented LFW dataset

Noise σ	Noisy data	FFDNET	DnCNN
30	**0.8165**	0.7984	0.8036
60	0.7467	0.7881	**0.7984**
90	0.7028	0.7855	**0.8062**

Bold value indicates highest result with respective of different noise level

Table 5 Result of Viola–Jones on output of image denoising algorithm on augmented FDDB dataset

Noise σ	Noisy data	FFDNET	DnCNN
15	**4852**	4781	4763
30	4503	**4730**	4691
45	4027	**4633**	4622

Bold value indicates highest result with respective of different noise level

Table 6 Comparison of Viola–Jones results on FDDB dataset with denoising algorithms

	Detected faces count
Original	**4815**
FFDNet	4800
DnCNN	4808

Bold value indicates highest result with respective of different noise level

Table 6 gives that we can detect 4815 faces out of a total of 5170 faces. Among denoising algorithms, DnCNN [9] outputs can detect the most faces, but they cannot outperform the original result. As a result, we can conclude that utilizing denoising methods does not result in a substantial improvement in face detection for a raw dataset.

We were unable to demonstrate that denoising techniques enhance substantial outcomes in face detection, as we were unable to do in face recognition. So, we ran an experiment on the augmented dataset, and it is clear that the results are considerably better. According to the table, FFDNet [8] outperforms DnCNN [9] for all three sigma levels.

As given in Table 7, DnCNN [9] can outperform the original dataset for object detection. As a result, we did not conduct any additional experiments with the augmented dataset.

Based on various metrics, we can conclude that DIP [10] works best for SSIM [11] but has no best impact for improving computer vision applications. In terms of standard bench marking PSNR [11], FFDNet [8] produces the best results. FFDNet [8] offers superior performance for some applications. DnCNN [9] produces the best results in terms of BRISQUE [12], and DnCNN [8] produces the best results in terms of the remaining applications.

Table 7 mAP value comparison for object detection on denoising algorithm

	mAP
Original	0.795
FFDNet	0.800
DnCNN	**0.807**
DIP	0.779

Bold value indicates highest result with respective of different noise level

7 Discussion and Conclusion

We have presented a comparison and analysis of three different image denoising methods. Results of image denoising are discussed on three standard metrics. Then, we have experimented with image denoising on CV applications. The image denoising methods are very effective when Gaussian noise is added to images. The performance of CV algorithms definitively improves when noisy images are denoised by deep learning algorithms. The performance of vision algorithms does not improve significantly when Gaussian noise is not added to images in the datasets. Quality of image has a high impact on the performance of CV algorithm which is proved by result on the noisy datasets, however, standard vision datasets have latent clean images and it is unlikely that performance will increase with significant change.

References

1. Fan L, Zhang F, Fan H et al (2019) Brief review of image denoising techniques. Vis Comput Ind Biomed Art 2:7
2. Turk MA, Pentland A (1991) Face recognition using eigenfaces. In: Proceedings of 1991 IEEE computer society conference on computer vision and pattern recognition, pp 586–591
3. Viola P, Jones M (2001) Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001
4. Redmon J, Divvala S, Girshick R, Farhadi A (2016) You only look once: unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition
5. Kumar P, Dick A, Sheng TS (2009) Real time target tracking with pan tilt zoom camera. Digital Image Comput Techn Appl 2009:492–497. https://doi.org/10.1109/DICTA.2009.84
6. Portilla J, Strela V, Wainwright MJ, Simoncelli EP (2003) Image denoising using scale mixtures of gaussians in the wavelet domain. IEEE Trans Image Process 12(11):1338–1351
7. Jain V, Learned-Miller E. FDDB: a benchmark for face detection in unconstrained settings. University of Massachusetts, Amherst
8. Zhang K, Zuo W, Zhang L (2017) FFDNet: toward a fast and flexible solution for CNN based image denoising. IEEE Trans Image Process
9. Zhang K, Zuo W, Chen Y, Meng D, Zhang L (2017) Beyond a Gaussian Denoiser: residual learning of deep CNN for image denoising. IEEE Trans Image Process 26(7)
10. Lempitsky V, Vedaldi A, Ulyanov D (2018) Deep image prior. IEEE/CVF Conf Comput Vis Pattern Recogn 2018:9446–9454. https://doi.org/10.1109/CVPR.2018.00984

11. Al-Najjar Y, Chen SD (2012) Comparison of image quality assessment: PSNR, HVS, SSIM, UIQI. Int J Sci Eng Res 3:1–5
12. Mittal A, Moorthy AK, Bovik AC (2011) Blind/referenceless image spatial quality evaluator (BRISQUE). In: 2011 conference record of the forty fifth asilomar conference on signals, systems and computers (ASILOMAR)
13. Jocher G, Stoken A, Borovec J, NanoCode012, Christopher STAN, Liu C, Laughing, tkianai, yxNONG, Hogan A, lorenzomammana, AlexWang1900, Hajek J, Diaconu L, Marc KY, oleg, wanghaoyang0106, Defretin Y, Lohia A, ml5ah, Milanko B, Fineran B, Khromov D, Yiwei D, Doug D, Ingham F, Frederik, Guilhen, Colmagro A, Ye H; Jacobsolawetz, Poznanski J, Fang J, Kim J, Doan K, Yu L (2021, Jan 5) ultralytics/yolov5: v4.0 - nn.SiLU() activations, weights biases logging, PyTorch hub integration (version v4.0). Zenodo
14. Beitzel SM, Jensen EC, Frieder O (2009) Mean average precision. In: Liu L, Özsu MT (eds) Encyclopedia of database systems. Springer, Boston
15. Santos T, de Souza, dos Santos Andreza L, Avila S (2019). Embrapa wine grape instance segmentation dataset—embrapa WGISD (version 1.0.0) [data set]. Zenodo
16. Ide H, Kurita T (2017) Improvement of learning for CNN with ReLU activation by sparse regularization. In: 2017 international joint conference on neural networks (IJCNN)
17. Ioffe S, Szegedy C (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. In: Proceddings of international conference on machine learning, pp 448–456
18. Dabov K, Foi A, Katkovnik V, Egiazarian K (2006) Image denoising with block-matching and 3D filtering. In: Proceedings of SPIE 6064, image processing: algorithms and systems, neural networks, and machine learning, vol 606414, 17 Feb 2006. https://doi.org/10.1117/12.643267
19. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: 2016 IEEE conference on computer vision and pattern recognition (CVPR)
20. Shah M, Kumar P (2021) Improved handling of motion blur for grape detection after deblurring. In: 2021 8th international conference on signal processing and integrated networks (SPIN), pp 949–954. https://doi.org/10.1109/SPIN52536.2021.9566112
21. Gary BH, Jain V, Learned-Miller E (2007) Unsupervised joint alignment of complex images. ICCV
22. Diederik PK, Adam JB (2015) A method for stochastic optimization. In: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015
23. Ruder S (2016) An overview of gradient descent optimization algorithms. ArXiv:1609.04747
24. Sammut C, Webb GI (2011) Mean squared error. In: Encyclopedia of machine learning. Springer, Boston, MA
25. Jolliffe I (2011) Principal component analysis. In: Lovric M (ed) International encyclopedia of statistical science. Springer, Berlin, Heidelberg
26. Deng J, Dong W, Socher R, Li L, Kai L, Li F-F (2009) ImageNet: a large-scale hierarchical image database. IEEE Conf Comput Vis Pattern Recogn 2009:248–255. https://doi.org/10.1109/CVPR.2009.5206848
27. Martin D, Fowlkes C, Tal D, Malik J. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proceedings of 8th international conference on computer vision
28. Cortes C, Vapnik V (1995) Support-vector networks. Machine Learn 20(3):273–297

Diabetic Retinopathy Classification Using Pixel-Level Lesion Segmentation

Harshil Gandhi, Koushik Agrawal, Urvi Oza, and Pankaj Kumar

Abstract Diabetic patients are at a high risk of developing Diabetic Retinopathy (DR). Due to the great success of deep learning, automated DR diagnosis has become a promising technique for the early detection and severity grading of Diabetic Retinopathy. DR classification is the process of classifying fundus image into five risk levels on the basis of severity of diabetes. In this paper, we propose a novel approach to classify DR severity levels. This is done by first segmenting each pixel into six different lesion types and then using different lesion regions present in the fundus images to classify the image into five severity levels of DR. Further, the model is optimized using unique pre-processing techniques like downsampling and image augmentation. We are working on the seg set of the FGADR dataset. Working on the seg set of the FGADR dataset, we report results of the proposed approach which outperforms the previous state-of-the-art methods. Diagnosis of DR can improve significantly by using these automated techniques in detecting lesion regions as well as detecting DR severity levels.

Keywords Diabetic retinopathy · Segmentation · Classification · Deep learning

H. Gandhi · K. Agrawal · U. Oza (✉) · P. Kumar
Dhirubhai Ambani Institute of Information and Communication
Technology (DAIICT), Gandhinagar, Gujarat, India
e-mail: 201921009@daiict.ac.in

H. Gandhi
e-mail: 201801026@daiict.ac.in

K. Agrawal
e-mail: 201801066@daiict.ac.in

P. Kumar
e-mail: pankaj_k@daiict.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_29

1 Introduction

One of the main causes of blindness in adults is diabetes. The retinal tissues swell up leading to their bursting and bleeding, ultimately leading to blurry vision and loss of eyesight. Early detection of DR helps in early prevention and better treatment. Diabetic Retinopathy is classified into two types on the basis of morphological changes in the fundus images: Non-proliferative Diabetic Retinopathy (NPDR) and Proliferative Diabetic Retinopathy (PDR). The NPDR and PDR features and risk levels are described in the following sections.

According to international protocol [1, 2], 5 stages of Diabetic Risk levels have been established, namely: (0) No retinopathy, (1) Mild non-proliferative DR (NPDR), (2) Moderate NPDR, (3) Severe NPDR and (4) Proliferative DR. This severity grading depends on the number and size of various lesions that we will discuss in the next section. We performed DR classification by analysing fundus images. Fundus images are serial photographs of the interior of one's eye through the pupil and includes the retina, posterior pole, fovea, optic disc and the macula. Different DR lesion features present in fundus images are discussed below.

1.1 DR Features

Non-proliferative Diabetic Retinopathy (NPDR) occurs as an early-stage DR retinal disease which shows the following symptoms:

- **Microaneurysms (MA)** are red spots with sharp margins whose size are less than 125μm. They are found in the macular regions and is one of the earliest signs of DR.
- **Haemorrhages (HM)** are larger spots whose size is somewhat greater than 125μm. Haemorrhages often occur due to leakages in the capillaries. Blot HM and superficial HM are most common type of Haemorrhages.
- **Hard exudates (EX)** appear to have sharp margins around bright yellow spots. This happens due to the plasma leakage that takes place in the macular region.
- **Soft exudates (SE)** are identified by white spots in the fundus images. This is because of the nerve fibres that get swollen. They are also commonly known as cotton wool spots.

Proliferative Diabetic Retinopathy (PDR) occurs when the DR Retinal disease is in an advanced stage. It shows the following symptoms:

- **Neovascularization (NV)** mainly occurs in the PDR stage. This is identified by new abnormal blood vessels at the optic disc.
- **Intra-retinal microvascular abnormalities (IRMA)**, also known as vitreous hemorrhage, appear when the abnormal retinal blood vessels proliferates or spread within or around the vitreous body.

1.2 DR Risk Levels

According to the Early Treatment Diabetic Retinopathy Study (ETDRS), the Diabetic Retinopathy (DR) risk levels are listed in Table 1 and visualized in the Fig. 1b. This table gives that the DR severity levels are directly dependent on the different lesions in the fundus images. Hence to classify the fundus images into these five severity levels, it is necessary to identify the lesion regions in the fundus images.

Table 1 5 DR risk levels: dependence of the DR risk levels on the lesion regions present in the retinal image [4]

DR risk level	Lesions
No DR	No lesions
Mild NPDR	Presence of MA
Moderate NPDR	Presence of MA and HM Presence of cotton woot spots and exudates
Severe NPDR	Any of the symptopms Venus beading in two quadrants Presence of MA and extensive HM in 4 quadrants
PDR	Neovascularization Presence of preretinal and vitreous HM

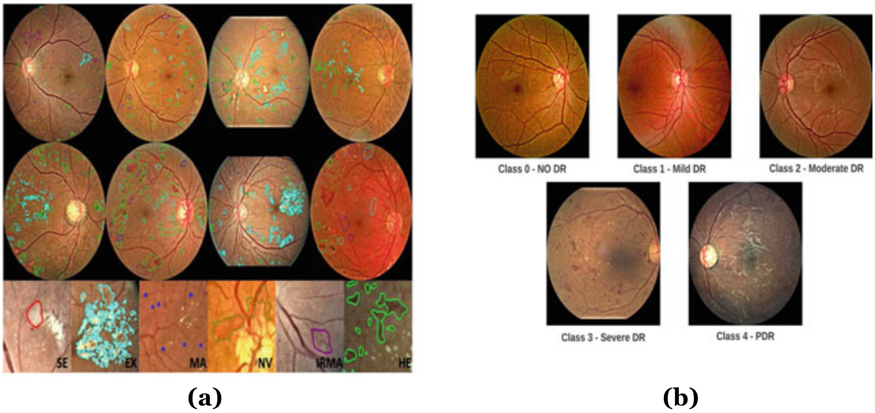


Fig. 1 Fundus images showing different lesion regions and different risk levels. **a** Examples of different lesion regions in images from the FGADR dataset images [3]. The six different colours represents six different lesion types. **b** Fundus images belonging to 5 stages of DR

2 Dataset

The FGADR dataset has two sets of data: the seg set and the grade set. The dataset we are using is the seg set from the FGADR [3] dataset. It consists of 1842 images with pixel-level lesion segmentations and image-level severity grading labels. The lesions segmented in the dataset include HE, MA, SE, EX, IRMA and NV.

At image level, the dataset is divided into 5 risks of DR: no DR, mild non-proliferative DR, moderate non-proliferative DR, severe non-Proliferative DR, proliferative DR.

Six lesions are in the form of 6 binary masks, where pixel value 1 represents pixels without any abnormalities/lesions and pixel value 0 represents pixels with that particular lesion. There are all 1842 masks available for the first four lesions, but only 159 masks for IRMA and 49 masks for NV.

Other DR lesion segmentation datasets available include the IDRiD dataset [5],the DRIVE dataset [6], the DDR dataset [7], e-opthaEX and e-opthaMA [8]. The IDRiD dataset contains 516 images, out of which only 81 of them have pixel-level lesion annotations in the form of binary masks. Only 4 abnormalities associated with DR are provided. The DRIVE dataset contains 40 images of pixel-level binary vessel masks. It is used for evaluating the segmentation of blood vessels in retinal images. The DDR dataset has 757 images with pixel-level lesion segmentation for only four lesion types. And finally, the e-opthaEX and e-opthaMA are two pixel-level segmented datasets with EX and MA abnormalities, respectively.

For our purpose, we need both pixel-level lesion masks as well as image-level DR severity levels. Among all of these datasets, only FGADR and IDRiD datasets have both of these. Considering the small dataset size of the IDRiD dataset (81 images with 4 lesions segmented), the FGADR dataset (1842 images with 6 lesions segmented) can be considered as the best choice for our purpose. Also, the images and the masks in the FGADR dataset are already cropped, i.e. the retina is concentrated to the centre of the image leaving no unnecessary black area in the sides of the dataset. However, all other datasets have to be cropped to remove the black area in the background. All the images are of the same dimensions 1280 × 1280, hence there is no need for padding to the dataset to make them the same dimensions.

3 Related Work

The problem of DR classification has been approached in one of the two ways in the past:

Firstly, taking the fundus image as input and directly classify them in one of the 5 risk levels using CNNs or SVM models. Deep learning (DL) and feature extraction models have been used for DR classification. In [9], firstly blood vessels were identified and then the retinal images were classified using SVM-based classifier model. It resulted in 80.4% accuracy. Patterns of symptoms such as exudates, haemorrhages, cotton spots and blood vessels are learnt and used for classification [10]. 30% sensitivity and 95% specificity and 75% accuracy were reported on 13 layered CNN which were trained on a small subset of Kaggle's Diabetic Retinopathy dataset and for a fewer epochs on entire dataset [11].

Secondly, similar to our method, segmenting the image into patches where there are some kind of abnormalities and then classify the images. As in Yang. et al. [12], more attention is given to lesion patches for DR grading using imbalance weighting map. In [13], attention maps were learnt by a zoom-in-net. This network shows lesion regions and then provides the severity levels of DR. These severity levels are shown at both local and global levels. In [14], image patches containing lesions were extracted from the Kaggle retinopathy dataset by 2 opthanmologists. These image patches were used to train a CNN using sliding window method and the five risk levels were predicted.

Binary classification (DR and No DR) is used in various research works wherein the first class is considered as No DR and the remaining four classes are considered as DR. Inception V3 network was used in [15] which employed binary classification on kaggle dataset resulting in accuracy of 90.9%. Models including VGG Nets, Alexnet, Xception, Resnet and Inception V3 were used for binary classification. With hyperparameter tuning, the highest accuracy was achieved by VggNets [16]. Multi-channel Inceptionv3 networks are proposed to perform binary classifications giving highest accuracy 85.2%, sensitivity 83.4% and specificity 87.6% [17]. We have even done binary classification to compare our results with some of these papers.

Just like ours, some of the approaches have tried to combine the two approaches, but in different ways. In our approach, we have given all six binary masks along with the original image as an input to the classification network and trained the network from scratch. In this paper, we propose unique pre-processing techniques for fundus images. Further, a novel network architecture is used for segmenting classifying fundus images into five stages of DR.

4 Proposed Architecture

Most of the work done in this domain have directly given the fundus images as an input to CNNs. These neural networks are like black boxes. We cannot supervize the mechanism of the models, and hence, we are unaware of the features that are extracted inside these networks and used for classification. We can speculate that the reason for lower accuracy in the prevalent works of DR classification is that the networks aren't able to extract the right feature required for the classification. The DR severity levels are directly dependent on the different lesions present in the fundus images as seen in Table 1. Hence, we propose to break down the classification task into two sub tasks. The first one being segmentation of the fundus images into six different abnormalities or lesions and the second one being classification of the

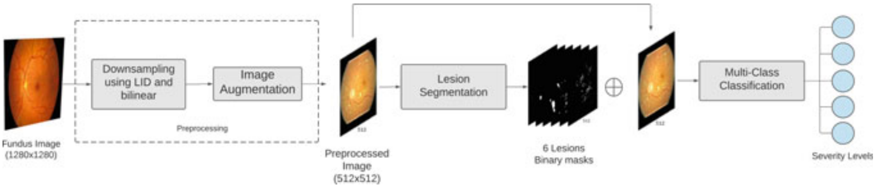


Fig. 2 Proposed architecture: pre-processing the images before giving it as an input to segmentation models. Then using the 6 segmented masks, along with the original image to classify the model into 5 severity levels

fundus images and the masks (obtained from the segmentation) into five severity levels of DR. This way, the network has a better idea about the features on which it should classify the image (Fig. 2).

4.1 Pre-processing

- **Downscaling**: The FGADR dataset contains images and mask for each lesion type of dimension 1280 × 1280. We downsample our images using Learned Image Downscaling(LID) [18]. LID is a adaptive downsampling algorithm that retains the original image's content when downsampled. LID trains a CNN to generate both downscaled and upscaled original size images.
  These pre-processed images (1280 × 1280) cannot be directly fed into CNN since the number of parameters and computations of CNN rises. Thus the images are downsized to 512 × 512 dimensions. Some of the lesions, such as MA are so minute that in order to make use of all of the features of the images, the images weren't downsized further. In the CNN architecture, downsized images (512 × 512) were given as an input and same sized masks as an output, to train the network.
- **Augmentation**: We have augmented the training data in order to obtain a model that is stable and does not overfit. This process is also necessary to cope with the problem of limited data available in the 4th and 5th (IRMA and NV) lesions. Augmentation makes the model more robust to real-life test scenarios. The augmentation operations that were performed are listed below.
  - Rotation: Images were rotated randomly between − 20°and 20°
  - Brightness and contrast: Brightness and contrast of images were randomly changed between -0.2 and 0.2
  - Vertical flip: Images were flipped randomly along its vertical axis
  - Horizontal flip: Images were flipped randomly along its horizontal axis.

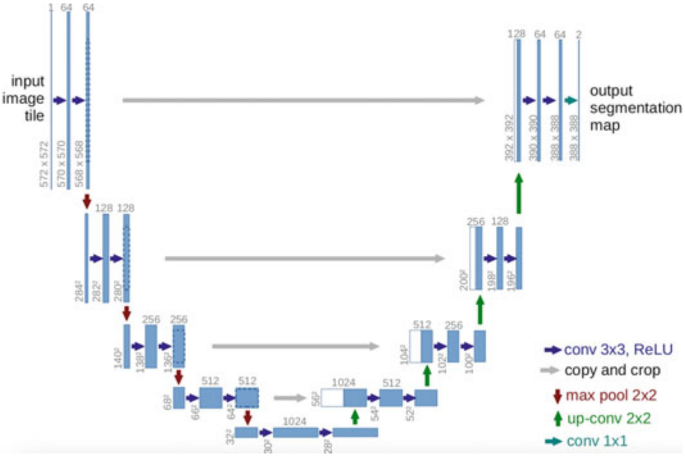


Fig. 3 Unet model [19] for segmenting images into different lesions

4.2 Segmentation

We attempt to find lesion regions in the images, using semantic segmentation methods of CNNs. We use Unet (with the backbone of densenet) network architectures to train our network.

Unet: Unet [19] is the extended version of an encoder–decoder fully convolutional network. The convolutional layers learn low and high-dimensional features as they iteratively train, using the filters in these layers to extract features. The idea behind Unet is to encode the image while it is downsampled by sending it through a CNN, then decode or upsample it to retrieve the segmentation mask. The learnt weight filters, upsampling and downsampling blocks (which can be made learnable), and concatenations and skip connections determine which characteristics are identified in the mask. The architecture of Unet is as shown in Fig. 3. The backbone is an architectural element that specifies how these layers are organized in the encoder network and how the decoder network should be constructed. The backbone used are often classic convolutional neural networks such as VGGNet, InceptionNet, EfficientNet, etc., which performs encoding and downsampling by itself. To create the final Unet, these networks are extracted and their equivalents are formed to execute decoding and upsampling. For our purpose we have taken densenet as the backbone for Unet architecture.

We are training six different networks for six different lesions, rather than training a single multi-class CNN architecture, mainly for 2 reasons:

- All six lesions are of a variety of different sizes, shapes, colours, etc., and our CNN models weren't able to capture the complexity required to classify all six different lesions. To train a multi-class semantic segmentation network, more complex networks were required, increasing the time and computation power.

- Some pixels might belong to more than one lesion because of overlapping of two different types of lesion regions. A multi-class segmentation requires one hot encoding for each pixel and while predicting, argmax function is used to determine the pixel class. Hence, each pixel won't be able to belong to more than 2 classes in a multi-class segmentation network.

We have performed the segmentation experiment using both models: a single multi-class CNN model and six binary CNN models. Both the results are compared in Sect. 5. All the models(segmentation and classification) are trained from scratch, without any pretrained weights.

4.3 Classification

This module aims to classify the image into five severity levels as discussed in Sect. 1.2. The predicted masks of the segmentation models, along with the original images, are given as an input to the classification models. Hence, nine channels (3 RGB + 6 masks) are given as an input to the networks. Networks used in training are InceptionNet v3 [20] and Densenet121 [21].

After performing the classification on five severity levels, the model is evaluated and results are recorded. We further consider class 0 as No DR and classes 1, 2, 3 and 4 as DR and measure the performance of our model on these two classes.

4.4 Evaluation Metric

For the segmentation part, we use two metrics:

- **IoU score (Jaccard coefficient)**: The region of overlap between the predicted segmentation and the ground truth divided by the area of union between them.

$$\text{IoU score} = \frac{\text{Area of Intersection}}{\text{Area of Union}} \tag{1}$$

- **F1 score (Dice coefficient)**: The harmonic mean of precision and recall gives $F1$ score. It keeps a balance between the two.

$$F1\,\text{score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{2}$$

For the classification of DR on the basis of its severity levels, we use accuracy as the main metric.

- **5 class Accuracy**: Accuracy is used to determine the proportion of fundus images that are accurately classified. A greater value of accuracy indicates correct classification of data. Accuracy is calculated using equation:

$$\text{Accuracy} = \frac{\text{Total correctly classified images}}{\text{Total test images}} \tag{3}$$

In binary classification, images can be divided into four categories on the basis of the actual class of the image and the predicted class of the image: true positive (TP), true negative (TN), false positive (FP) and false negative (FN). On the basis of these categories, we use three metrics to evaluate the model:

- **2 class Accuracy**: Accuracy is used to determine the proportion of fundus images that are accurately classified.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{4}$$

  A greater value of accuracy indicates correct classification of data.
- **Sensitivity**: Sensitivity measures what propotion of fundus images that are classified as having DR actually has DR.

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{5}$$

  A greater value of sensitivity indicates that model performed well in correctly detecting the disease in patients.
- **Specificity**: Specificity measures what proportion of fundus images that are classified as not having DR actually does not have DR.

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{6}$$

  A greater value of specificity indicates that the model performed well in detecting that there is no disease.

5 Results

The IoU score and $F1$ score are used to evaluate the segmentation model, as indicated in the previous section. Table 2 gives the image segmentation results performed on Unet (with the backbone of DenseNet). The model has performed efficiently on the data.

Table 2 Lesion segmentation results performed on Unet with the backbone of densenet for the six class of lesions

Lesion classes	IOU score	$F1$ score
Class 0 (MA)	0.363	0.505
Class 1 (HE)	0.421	0.584
Class 2 (EX)	0.471	0.630
Class 3 (SE)	0.305	0.456
Class 4 (IRMA)	0.498	0.653
Class 5 (NV)	0.468	0.631

Table 3 Classification results: comparision of results that were obtained with proposed architecture by other state-of-the-art methods

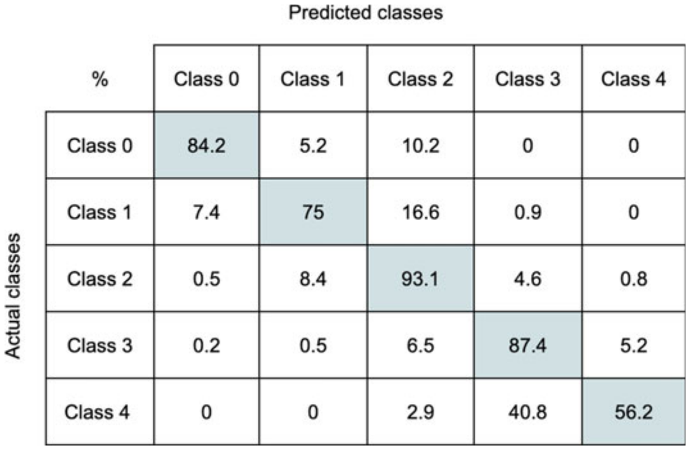
Models	5 class acc.	2 class acc. (%)	Sensitivity (%)	Specificity (%)
Densenet121	80.9%	97.0	80.7	97.9
InceptionNet V3	**81.0%**	**98.2**	**84.2**	**98.9**
Multi-channel architecture with LID [17]	NR	85.2	83.4	87.6
BT-VGG [22]	NR	83.2	81.8	88.3

(*NR* not reported). Bold represents experiments performed by us and best scores achieved

Table 3 gives the classification results. Accuracy is used as the main evaluating metric for five class classification, and accuracy, sensitivity and specificity are used for binary classification. The images were categorized into five severity levels using InceptionNet v3 and Dense Net121. The results are compared with multi-channel architecture with LID [17] and the BT-VGG [22].

Five classes of DR are the severity levels on the scale of 0–4 in the increasing order of DR severity. It is very difficult for even professional ophthalmologists to exactly classify the images into the five severity levels. Ideally, the model should classify the images correctly. The second best scenario in DR grading is to have a prediction near the actual severity level. Hence in DR grading, even when the images are not correctly classified, it is better to have the image classified into a class which is adjacent to the actual class. Figure 4 shows the confusion matrix for the InceptionNet model results. The confusion matrix indicates that the majority of the images that are incorrectly classified, belong to severity levels adjacent to the actual severity level.

Furthermore, when lesion masks are provided, none of the grade-4 DR images are incorrectly predicted as Grade-0 or Grade-1. These enhancements make DR diagnosis systems more stable and understandable for ophthalmologists since there is no misclassification from high-severity DR levels to regular or early-stage DR levels. When the lesion masks are provided, neither of the grade-0 DR images are incorrectly rated as Grade-3 or Grade-4. This shows the high efficiency of this method
Table 4 compares the average $F1$ score for both multi-class segmentation and binary segmentation. This score is the average of the $F1$ scores of all 6 lesions present. The



Predicted classes

Actual classes

%	Class 0	Class 1	Class 2	Class 3	Class 4
Class 0	84.2	5.2	10.2	0	0
Class 1	7.4	75	16.6	0.9	0
Class 2	0.5	8.4	93.1	4.6	0.8
Class 3	0.2	0.5	6.5	87.4	5.2
Class 4	0	0	2.9	40.8	56.2

Fig. 4 Confusion matrix of five class InceptionNetv3 classification results

Table 4 Comparing results of multi-class segmentation and binary segmentation for six lesion classes

Models	Average $F1$ score
1 multi-class segmentation model	0.481
6 binary segmentation models	0.577

table gives that it is beneficial to train a binary segmentation network 6 times (for 6 lesions), than to train a single multi-class segmentation network. Reasons for this are already discussed in Sect. 4.2.

6 Conclusion

Giving only fundus images as input to the CNNs makes it difficult for the model to learn the required features to classify images. If the lesion regions are identified first and then given as an input along with the original fundus images, the machine is aware of what features are to be extracted and what features are essential for the different severity levels of DR. Our approach gives significant results and hence we conclude that it is better to divide the task into two individual sub tasks, segmentation and classification.

Further, the segmentation model stand alone has performed quite well and can be used to assist doctors while diagnosing. Unique pre-processing techniques have further helped the model improve its accuracy. Augmenting the images has helped in dealing with the limited data available and also provides robustness to our model so that it can deal with test images in a better way. Also, downscaling the images using LID method have helped preserve lesion features even in the smaller-sized images.

Further, it is beneficial to train a binary segmentation network 6 times (for 6 lesions), than to train a single multi-class segmentation network. Thus, we propose a newer approach to classify fundus images into distinct severity levels. Computer-based DR grading can be used in future as an effective alternative to the traditional manual analysis of fundus images.

References

1. Gulshan V, Peng L, Coram M, Stumpe MC, Wu D, Narayanaswamy A, Venugopalan S, Widner K, Madams T, Cuadros J, Kim R, Raman R, Nelson PC, Mega JL, Webster DR (2016) Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. JAMA 316(22):2402. https://doi.org/10.1001/jama.2016.17216
2. Ophthalmoscopy D, Levels ETDRS (2002) International clinical diabetic retinopathy disease severity scale detailed table
3. Zhou Y, Wang B, Huang L, Cui S, Shao L (2021) A benchmark for studying diabetic retinopathy: segmentation, grading, and transferability. IEEE Trans Med Imaging 40(3):818–828. https://doi.org/10.1109/tmi.2020.3037771
4. Srinivasan V, Bhanu V (2020) A review on diabetic retinopathy disease detection and classification using image processing techniques 7(9):546–555
5. Porwal P, Pachade S, Kamble R, Kokare M, Deshmukh G, Sahasrabuddhe V, Meriaudeau F (2018) Indian diabetic retinopathy image dataset (IDRiD): a database for diabetic retinopathy screening research. Data 3(3):25. https://doi.org/10.3390/data3030025
6. Staal J, Abramoff MD, Niemeijer M, Viergever MA, van Ginneken B (2004) Ridge-based vessel segmentation in color images of the retina. IEEE Trans Med Imaging 23(4):501–509. https://doi.org/10.1109/tmi.2004.825627
7. Li T, Gao Y, Wang K, Guo S, Liu H, Kang H (2019) Diagnostic assessment of deep learning algorithms for diabetic retinopathy screening. Inf Sci 501:511–522. https://doi.org/10.1016/j.ins.2019.06.011
8. Decencière E, Cazuguel G, Zhang X, Thibault G, Klein J-C, Meyer F, Marcotegui B, Quellec G, Lamard M, Danno R, Elie D, Massin P, Viktor Z, Erginay A, Laÿ B, Chabouis A (2013) TeleOphta: machine learning and image processing methods for teleophthalmology. IRBM 34(2):196–203. https://doi.org/10.1016/j.irbm.2013.01.010
9. Carrera EV, González A, Carrera RA (2017) Automated detection of diabetic retinopathy using SVM. In: 2017 IEEE XXIV international conference on electronics. Electr Eng Comput (INTERCON):1–4
10. Faust O, Acharya UR, Ng EY, Ng KH, Suri JS (2010) Algorithms for the automated detection of diabetic retinopathy using digital fundus images: a review. J Med Syst 36(1):145–157. https://doi.org/10.1007/s10916-010-9454-7
11. Pratt H, Coenen F, Broadbent DM, Harding SP, Zheng Y (2016) Convolutional neural networks for diabetic retinopathy. Proc Comput Sci 90:200–205. https://doi.org/10.1016/j.procs.2016.07.014
12. Yang Y, Li T, Li W, Wu H, Fan W, Zhang W (2017) Lesion detection and grading of diabetic retinopathy via two-stages deep convolutional neural networks. ArXiv:1705.00771
13. Wang Z, Yin Y, Shi J, Fang W, Li H, Wang X (2017) Zoom-in-Net: deep mining lesions for diabetic retinopathy detection. ArXiv:1706.04372
14. Lam C, Yu C, Huang L, Rubin D (2018) Retinal lesion detection with deep learning using image patches. Investigative Opthalmol Vis Sci 59(1):590. https://doi.org/10.1167/iovs.17-22721

15. Hagos MT, Kant S (2019) Transfer learning based detection of diabetic retinopathy from small dataset. ArXiv:1905.07203
16. Wan S, Liang Y, Zhang Y (2018) Deep convolutional neural networks for diabetic retinopathy detection by image classification. Comput Electr Eng 72:274–282. https://doi.org/10.1016/j.compeleceng.2018.07.042
17. Doshi N, Oza U, Kumar P (2020) Diabetic retinopathy classification using downscaling algorithms and deep learning. In: 2020 7th international conference on signal processing and integrated networks (SPIN), pp 950–955
18. Sun W, Chen Z (2020) Learned image downscaling for upscaling using content adaptive resampler. In: IEEE transactions on image processing: a publication of the IEEE signal processing society. https://doi.org/10.1109/TIP.2020.2970248 Advance online publication
19. Ronneberger O, Fischer P, Brox T (2015) U-Net: convolutional networks for biomedical image segmentation. In: Navab N, Hornegger J, Wells WM, Frangi AF (eds) Medical image computing and computer-assisted intervention—MICCAI 2015: 18th international conference, Munich, Germany, 5–9 Oct 2015, proceedings, part III. Springer International Publishing, pp 234–241. ISBN: 978-3-319-24574-4
20. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. IEEE Conf Comput Vis Pattern Recogn (CVPR) 2015:1–9. https://doi.org/10.1109/CVPR.2015.7298594
21. Huang G, Liu Z, Pleiss G, Van Der Maaten L, Weinberger K (2019) Convolutional networks with dense connectivity. IEEE Trans Pattern Anal Mach Intell:1. https://doi.org/10.1109/tpami.2019.2918284
22. Adly MM, Ghoneim AS, Youssif AA (2019) On the grading of diabetic retinopathies using a binary-tree-based multiclass classifier of cnns. Int J Comput Sci Inf Secur (IJCSIS) 17:109–114

Password-Protected Lossless Visible Watermarking Technique for Digital Image

Lalita Kumari, Pradeep Kumar Singh, and Munesh Chandra

Abstract Sharing digital images through social networking and other means of the Internet has placed its ownership and copyright at the corner. Protection from unauthorized use of digital images is gradually becoming a challenging task. Digital watermarking is used as the shield against data theft and unauthorized use of digital images without the consent of their owner. In this paper, we propose a password-protected visible watermarking system (PPVWS) to protect against its use without proper citation or proper permission from the owner of the protected image. This method describes an effective way to watermark a visible image or logo to any digital image so that the watermarked image reflects the source/owner of the image. The watermarked image produced in this technique is password protected, and the original image is self-constructed from the watermarked image when the correct password is produced. The proposed algorithm is able to embed the watermark through a password, provided by the owner. The watermark removal algorithm, presented in this paper, is performed through the same password that was given at the time of embedding the watermark. The proposed technique has been shown to secure enough for cryptanalysis as well as bruit force.

Keywords Watermarking · Multimedia · LSB · PPVWS · IMRM · Password-protected watermarking · Key-based watermarking

L. Kumari
Amity University Patna, Patna, India

M. Chandra
NIT Agartala, Agartala, India

P. K. Singh (✉)
Narsee Monjee Institute of Management Studies, School of Technology Management and Engineering (STME), Chandigarh, India
e-mail: pradeep_84cs@yahoo.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_30

1 Introduction

The present era is dedicated to Internet technology in which mobile phones, computers, laptops, tablets are the key point for communication. Digital images are the most common and efficient way of presentation and communication. Digital images play a vital role to share something in an illustrative way. Everyone is using/reproducing the images intentionally or unintentionally through social networking or other means of communication, which may be an asset of someone else. Like other assets, digital images are also a digital asset, and it must be protected from piracy. But, the reproduction of others image without proper permission/citation is in common practice, and it must be restricted. The solution of this problem is visible watermarking, where some image or text, representing the owner of the image, are embedded with the original image such that these two images combine to one non-separable image. Image in such a way is visible, but it displays some visible data/logo related to the owner or generator of the image. This way owner of the image can share it through different means of communication with embedded watermarked logo. And if some other person is using this image, the watermarked citation also has to show. For a good watermarking technique, the watermark must be extracted to reconstruct the original image by the legitimate user with a good value of SNR and PSNR. A good watermarking technique must also be secured from brute force and cryptanalysis attacks.

Digital watermarking is used for time stamping, authentication, and protection of copyright. The digital watermarking technique is divided into two classes: invisible watermarking and visible watermarking [1]. Visible watermarking can perform copyright protection in a more direct and immediate manner than invisible watermarking [2]. Invisible watermarking is only for validation and does not show any watermark on the image, so the image may be reproduced by bypassing the authenticity, but in visible watermarking, same is not possible. The digital library, e-commerce, digital press, social networking sites are the main applications of visible watermarking [3–5]. These digitally watermarked images may be set free for reading, demo, or other free uses purpose in limited conditions. Protection of digital images by adding some translucent logo/image/text is very effective on a copy of the original image as watermarked content indicates originality and the source of the image. Concerning watermark robustness, the visible watermarking inherently provides robustness against a wide range of attacks; the watermarked image has received several attacks, such as geometrical attacks including scaling, rotation, transformation, and signal processing attacks consisting of compression, filtering, noise addition, and modification of brightness and contrast, among others [6, 7]. In this paper, we have proposed a visible watermarking method in which watermark can be added to original image. Embedding of watermark is password protected which is essentially required to remove watermark from the original image.

2 Literature Survey

Watermarking method started to evolve for the protection of printed documents in 1282 in Italy [1] which gradually improved to digital image watermarking since 1995 [8]. Over the decades of research in the field of digital watermarking, a large number of research algorithms have been proposed which classified [1, 8] as presented in Fig. 1. Most of the methods are for invisible watermarking, and only, a few are presented for visible watermarking. These methods are further classified as visible and invisible watermarking. Methods for invisible watermarking cannot be extended to be used for visible watermarking. The invisible watermarking technique hides the watermark in the original image and is used to establish authenticity and verification. Visible watermarking does not hide the watermark; instead, it displays inside the original image to display copyright/proprietary information.

Visible watermarking of the monochrome image is presented in [4, 9] which compresses the portion of the original image and embeds payload/watermark image into the original image [10]. Watermarking of image just after capturing from a single-sensor digital camera is presented in [2] which embed the watermark into Bayer color filter array (CFA) domain. But, it is used just after capturing the photo through a digital camera and does not apply to other digital images created from other means. Similar to the method in [4, 11], this is also non-reversible, i.e., no method proposed to remove the embedded watermark, whereas methods in [3, 7, 12–20] are removable/reversible. In this paper, we have presented a password-protected visible watermarking system (PPVWS) which can embed binary, gray, and color images as a watermark on base/original image. It is useful to show visible information about ownership of the digital image. This method presents a technique to add a secret password as key at the time embedding the watermark which is again required at the time of removing the watermark. Figure 2 represents the proposed PPVWS framework.

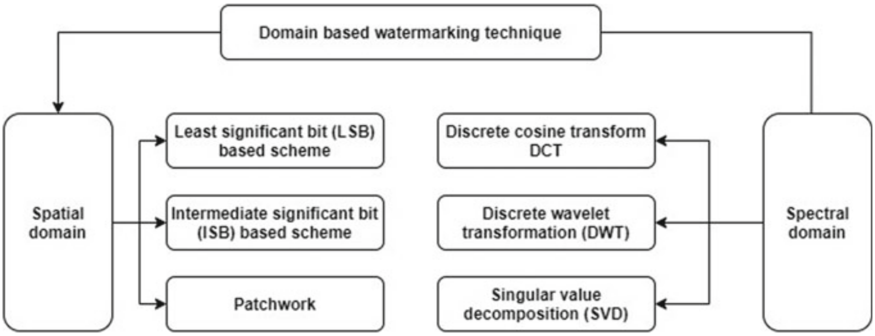

Domain based watermarking technique
Spatial domain
Least significant bit (LSB) based scheme
Intermediate significant bit (ISB) based scheme
Patchwork
Discrete cosine transform DCT
Discrete wavelet transformation (DWT)
Singular value decomposition (SVD)
Spectral domain

Fig. 1 Classification of digital image watermarking

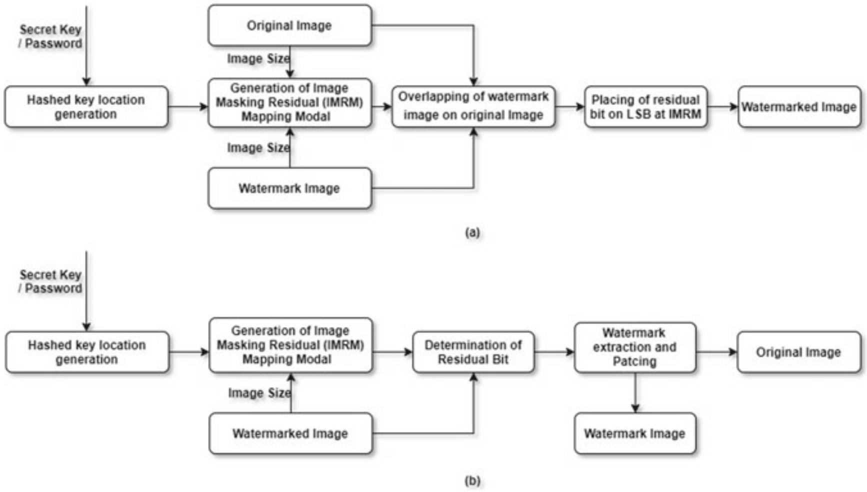

Secret Key / Password
Hashed key location generation
Original Image
Image Size
Generation of Image Masking Residual (IMRM) Mapping Modal
Image Size
Watermark Image
Overlapping of watermark image on original Image
Placing of residual bit on LSB at IMRM
Watermarked Image
(a)
Secret Key / Password
Hashed key location generation
Generation of Image Masking Residual (IMRM) Mapping Modal
Image Size
Watermarked Image
Determination of Residual Bit
Watermark extraction and Patcing
Original Image
Watermark Image
(b)

Fig. 2 Framework for PPVWS **a** watermark embedding **b** watermark removal

3 Proposed Method

Proposed PPVWS performs on the color base image, with a colored watermark image. The watermark image is placed at the center of the base image. The residual bit values are determined through differences in original pixels value and generated pixel value and are distributed over the height width of the base image. The distribution of residual bits is controlled by pseudorandom key location generated through a user-provided password. Watermarking is done in PPVWS framework in such a manner that the original image is extracted through the same framework if the correct password is provided. The proposed method is designed to address the following points: (1) Applicable on the color base image and colored watermark logo. (2) Visible watermark in all background. (3) Self-sufficient watermarking for lossless recovery of an original image by a valid user through the correct password. (4) Satisfactory level of protection against security threats. The proposed method is divided into the following parts:

3.1 Partitioning of Base Image into Blocks

The image to be watermarked is generally kept small in comparison to the original base image. This small size watermark gives an advantage of storing metadata at a hashed location of the original image (image to be watermarked). The original base image is divided into 9 blocks; out of which, one block is used for watermarking, while the rest 8 blocks are used to keep additional information. This additional

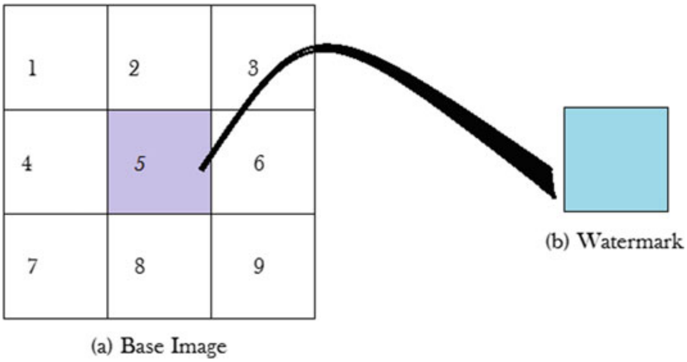


Fig. 3 Mapping of watermark block with image block

information is used to retrieve the original image by a valid user through the correct password. As shown in Fig. 3, the proposed method places the watermark at the center of the base image, i.e., block-5, which may be repositioned to another block as per the requirement of watermark placement position.

3.2 Embedding of Watermark on Base Image

Embedding of watermark on the base image is done through superposition of watermark on the center-aligned desired block of the base image, as shown in Fig. 4. A pixel value of the embedded image in the target block is represented by Eqs. (1) and (2). Here, *Ib*(*i*, *j*) is a pixel value of input base image, and *Im*(*i*, *j*) is pixel value of watermark at location *i*, *j* target block where *i*, *j* ∈ domain of target block in base image translated to pixel location (0, 0). Alpha is the intensity weight constant being applied on the watermark.

$$\delta I_{(i,j)} = \left| Ib_{(i,j)} - Im_{(i,j)} \right| \tag{1}$$

$$Iw'_{(i,j)} = Ib_{(i,j)} + \delta I_{(i,j)}.\alpha \tag{2}$$

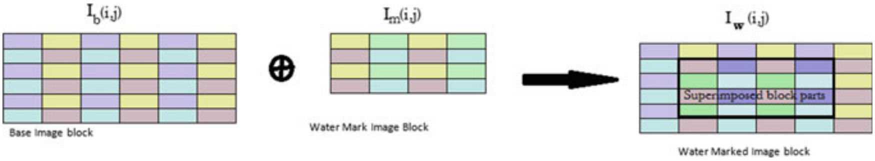


Fig. 4 Superimposition of watermark on desired location

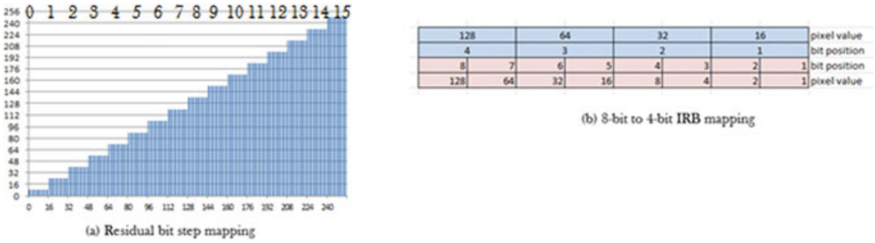
3.3 *Image Masking Residual Model (IMRM)*

The superimposed watermarked image block, generated in the previous section, is the target image that is supposed to be put in the public domain so that its ownership gets preserved by showing a non-removable desired watermark on it. But, the target of this presented algorithm is to put the watermark in such a way that it may be removed by the owner of the image. For this, original pixel values are kept by computing the difference between watermarked image bits and original image bits by down sampling through Eq. (3). Here, for 8-bit to 4-bit, down sampling value of η is 4. Figure 5 shows residual bit step size and mapping of 8-bit pixel value to 4-bit pixel values in Fig. 5a, b, respectively. Here, IRB is neglecting η LSB to get 8-bit to 4-bit mapping with minimum distortion. Numbers are shown at top of Fig. 5a is down-sampled pixel value corresponding to original pixel value shown on the bottom of Fig. 5a. The pseudorandom algorithm is presented below for residual bit placement.

$$I_R = \frac{I_w - I_b}{2^{\eta}} \tag{3}$$

Pseudo Algorithm For Residual Bit Placement

```
Step-1. Identify watermark placement block, $B_w$
Step-2. Embed watermark using Eq. (1) and Eq. (2)
Step-3. Compute residual bit, $I_{R,}$   from watermarked image
        and original image using Eq. (3)
Step-4. Convert signed $I_{R(i,j)}$ value to 5 bits data $b_4$, $b_3$,
        $b_2$, $b_1$, $b_0$ where $b_4$ is sign bit and $b_0$ to $b_3$ is data
        bit
Step-5. Put all b values for all the pixels within $B_w$ into
        a linear array.
Step-6. Randomize these residual bits through HLA
        discussed in Sect. 3.4
Step-7. Put these randomized HLA into LSB of pixel
        locations in non-watermarked blocks.
```



(a) Residual bit step mapping

128		64		32		16		pixel value
4		3		2		1		bit position
8	7	6	5	4	3	2	1	bit position
128	64	32	16	8	4	2	1	pixel value

(b) 8-bit to 4-bit IRB mapping

Fig. 5 Image residual bit (IRB) mapping: 8-bit to 4-bit

3.4 *Secret Key-Based Pseudorandom Hashed Location Array (HLA) Generation*

The IMRM takes the watermark image and base image as input and produce residual bits as output. These residual bits are generated as a sequential list. These sequential bits are now randomized for non-recovery of original sequences by the non-legitimate user. For this, a key-based hashed value is generated, and bits of this hashed key are used to produce HLA. This noiseless hashed location array (HLA) is secure enough to be generated without the actual key as well as 100 percent recovery of original sequential residual bits through the original key value.

3.5 *Storage of Residual Data on LSB Through HLA*

Once the HLA is generated, residual bits are stored at the LSB of these pixels as sequential data. Metadata are also stored in LSBs to keep track of block size, masking block, etc., which is used at the time of decryption. The generated HLAs are key based; unit original key is not determined; the HLA is not recoverable, resulting in non-recovery if residual bits which is used to the extract base image and watermark from the watermarked image.

4 Result and Discussion

The presented PPVWS is a novel lossless approach to watermarking an image into a base image in such a way that the watermark is visible and inseparable through bruit force/cryptanalysis. This watermarking is guided through a user-provided key/password which is essentially required for removal of the watermark that was initially set at the time of the watermarking.

Testing has been performed on colored base images of 512X512 pixels, on which the watermark image is also a colored image of 116X134 pixels. A grayscale watermark has also been used for the same. A grayscale watermark has only one layer with contrast to color, however, the color watermark has three layers, and their respective values of; MSE and SNR are smaller for the grayscale as compared to the colored watermark.

Embedding of color watermark on the fifth block of base image with different values of transparency coefficient ($0 < \alpha < 1$) is presented in Fig. 6. Figure 6a shows sample base image (fruits) with virtual partitioning into 3 × 3 equisized blocks on which embedding of watermark on central block (5th block) is to be performed, whereas Fig. 6b shows an a colored input watermark image and Fig. 6c–i shown

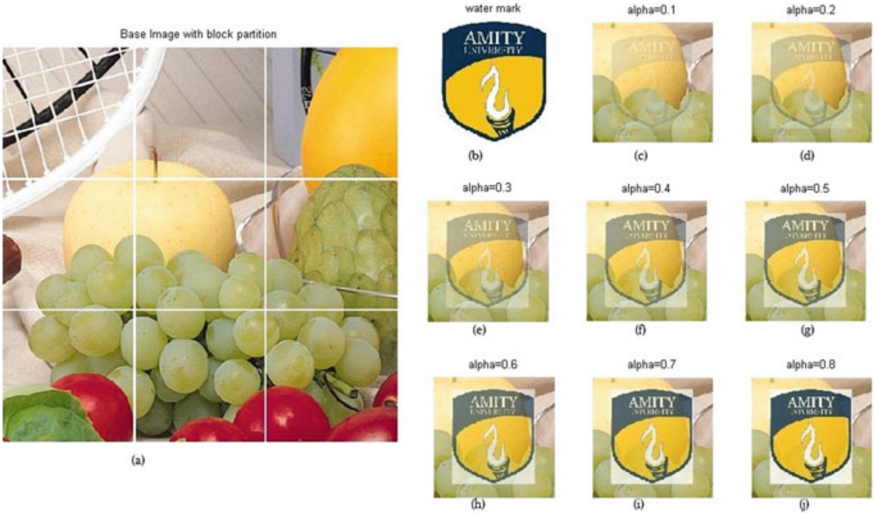


Fig. 6 Watermarking with different values of alpha on 5th block after partitioning base image into 3 × 3 blocks. **a** base image. **b** watermark image. **c** to **f** watermarked image with $\alpha = 01$ to 0.9, respectively

output of embedded 5th block of input image with value of transparency coefficient $\alpha = 0.1$ to 0.8, respectively. It shows step-1 and step-2 of the above presented pseudo-algorithm. Output of step-3 of the presented pseudo-algorithm is shown in Fig. 7, in which Fig. 7a is input base image (peppers) of size 512 × 512 pixel. Watermarked image with $\alpha = 0.3$ is shown in Fig. 7b, whereas Fig. 7c is the computed difference image block between original image block (Fig. 7a) and watermarked image block (Fig. 7b). At the time of computing difference, taking in account, instead a difference image block sign marker is generated separately which is of same size as that of the 5th block (watermarked block). This sign image block is shown in Fig. 7d. Fig. 7e–g shows the computed IMRM from the Fig. 7d with $\eta = 2, 4, 8$, respectively.

Based on the computed IMRM for different values of η, watermarked block is recomputed, and that modified watermarked block is shown in Fig. 8. Input base image block has been shown in Fig. 8a, whereas Fig. 8b is initial watermarked output image with $\eta = 1$ and $\alpha = 0.3$. IMRM has not been applied on Fig. 8b. Watermarked output image block on application of IMRM with $\eta = 2$, $\eta = 4$, and $\eta = 8$ has been shown in Fig. 8c–e, respectively. Figure 8g–j represents pixel value range display graph for all the three color components in the target block after application of IMRM model with $\eta = 1, 2, 4$, and 8, respectively. Figure 8f represents initial pixel value range distribution of all the three color components of target block in the base image.

A series of test has been applied on different input base images, and efficiency of presented algorithm has been measured using computation and comparison of MSE,

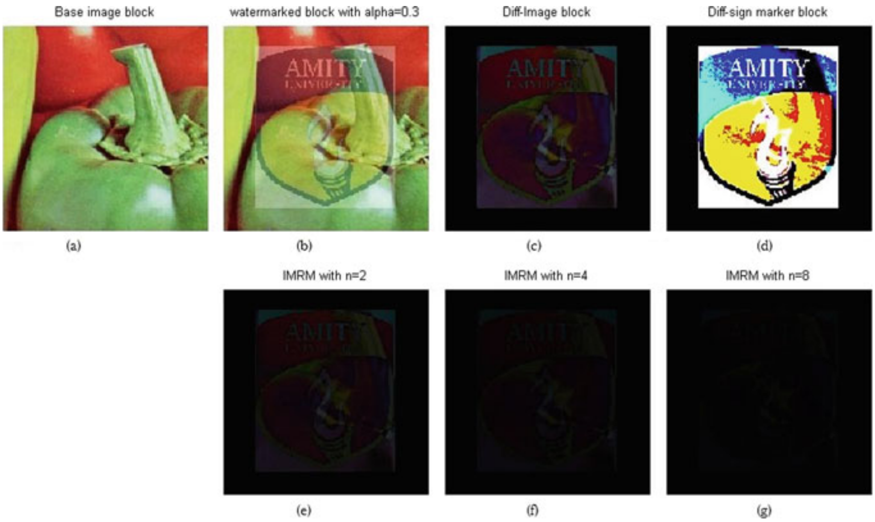


Fig. 7 Diff-image computation of 5th block on watermarking with alpha = 0.3. **a** Base image block, **b** watermarked image block with alpha = 0.3, **c** difference image computation, **d** diff-sign marker image for difference computation, **e** IMRM computation on $n = 2$, **f** IMRM computation on $n = 4$, and **g** IMRM computation on $n = 8$

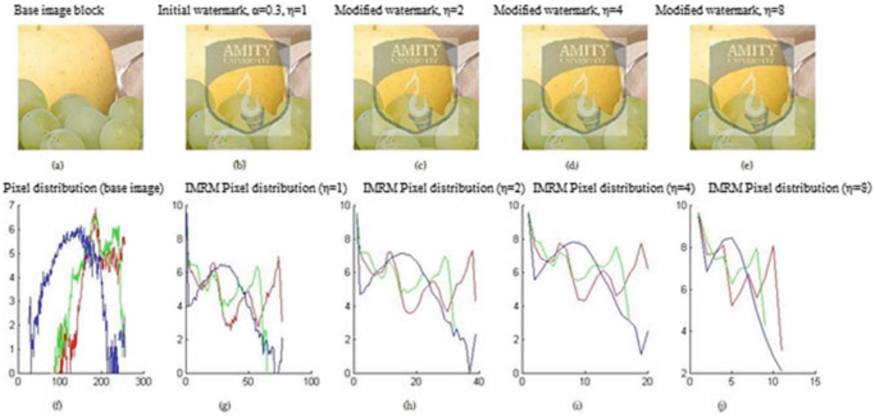


Fig. 8 Modified watermark image computation through IMRM (with different values of η). **a** base image block, **b** initial watermark with $\eta = 1$, **c** modified watermark from IMRM with $\eta = 2$, **d** modified watermark from IMRM with $\eta = 4$, **e** modified watermark from IMRM with $\eta = 8$, **f** pixel range display graph of base image. **g**–**j** pixel range display graph of MRSM for $\eta = 1, 2, 4, 8$, respectively

PSNR, and SNR. The method has been evaluated with different set of η and α. The evaluated performance measure has been summarized into Table 1. The presented method is so efficient that after embedding of watermark and storing of metadata at MSB of hashed series of location does not give any visual distortion. Watermarked output for input image fruits, baboon, lena, peppers has been shown in Fig. 9 where $\eta = 8$ and $\alpha = 0.3$.

5 Conclusion

In this paper, a new method has been presented which watermark is embedded at anyone on the 9 blocks obtained after dividing the base image into 3 × 3 equal blocks. The presented method stores metadata at MSBs of the rest 8 blocks in such a way that it can be regenerated the original base image if and only if the secret key is provided which was set at the time of embedding the watermark. The unique IMRM has been presented which shrinks the required bit space for storing mete information for removing the watermark. The pixel value of the difference image, (which is essentially required for watermark removal), ranges from 0 to 255. A positive or negative sign image marker is also obtained which is of the same size as the target block (one of the obtained 9 blocks). For $\eta = 2$, 4, and 8, the pixel value ranges from 0–127, 0–64, and 0–32 which on result requires 7, 6, and 5 bits, respectively. One addition bit is required for the sign bit marker; therefore finally, 8 bits, 7 bits, and 6 bits are required to store the additional information at a secret hashed location. The presented method divides the whole base image into 9 blocks on which one block is used for watermarking. The rest 8 locations are utilized to store maximum 8 bits at MSB (one at each location) which confirms the usability of the presented method. If $\eta = 1$ is set, the $8 + 1 = 9$ bit location is required that is not available. Therefore, the presented method proves its usability for $\eta = 2$, 4, and 8. The same is supported by the result Table 1. This table represents mean square error (MSE), signal-to-noise ratio (SNR), and peak signal-to-noise ratio (PSNR) for different value of η. Here, I_{Base} is the original image; IW is the watermarked image, whereas $\text{IW}_{\text{Expected}}$ is watermarked output image without IMRM, and $\text{IW}_{\text{obtained}}$ is watermarked output image with IMRM. $\text{IW}_{\text{obtained}}$ is the actual watermarked output image which is self-constructive to original image on production of secret key.

Table 1 Additive noise due to storing metadata for noiseless watermark recovery algorithm on different value of η

Base image		$\eta = 2$			$\eta = 4$			$\eta = 8$		
		MSE	SNR	PSNR	MSE	SNR	PSNR	MSE	SNR	PSNR
Lena	$\mathbf{I_{Base}}$ **&** $\mathbf{IW_{Expected}}$	60.556880	25.171606	30.309169	60.534973	25.173178	30.310740	60.981608	25.141253	30.278815
	$\mathbf{I_{Base}}$ **&** $\mathbf{IW_{Obtained}}$	60.968119	25.142213	30.279776	60.946213	25.143774	30.281336	61.384711	25.112639	30.250201
	$\mathbf{IW_{Expected}}$ **&** $\mathbf{IW_{Obtained}}$	0.411240	46.854014	51.989854	0.411240	46.854059	51.989854	0.403103	46.940782	52.076645
Fruits	$\mathbf{I_{Base}}$ **&** $\mathbf{IW_{Expected}}$	71.291734	25.948636	29.600412	71.239319	25.951830	29.603606	71.027588	25.964757	29.616533
	$\mathbf{I_{Base}}$ **&** $\mathbf{IW_{Obtained}}$	71.632848	25.927905	29.579681	71.580433	25.931084	29.582860	71.355474	25.944755	29.596531
	$\mathbf{IW_{Expected}}$ **&** $\mathbf{IW_{Obtained}}$	0.341114	49.106731	52.801808	0.341114	49.106791	52.801808	0.327886	49.279016	52.973576
Peppers	$\mathbf{I_{Base}}$ **&** $\mathbf{IW_{Expected}}$	61.760427	24.299469	29.587864	61.792969	24.297182	29.585576	61.988851	24.283436	29.571831
	$\mathbf{I_{Base}}$ **&** $\mathbf{IW_{Obtained}}$	62.264814	24.264145	29.552540	62.297356	24.261876	29.550271	62.434460	24.252329	29.540723
	$\mathbf{IW_{Expected}}$ **&** $\mathbf{IW_{Obtained}}$	0.504387	45.194927	50.503901	0.504387	45.194941	50.467329	0.445609	45.733000	51.005430
Baboon	$\mathbf{I_{Base}}$ **&** $\mathbf{IW_{Expected}}$	70.715012	24.326225	29.635687	70.777629	24.322381	29.631844	71.565755	24.274288	29.583751
	$\mathbf{I_{Base}}$ **&** $\mathbf{IW_{Obtained}}$	71.172048	24.298246	29.607709	71.234665	24.294427	29.603890	72.021491	24.246720	29.556183
	$\mathbf{IW_{Expected}}$ **&** $\mathbf{IW_{Obtained}}$	0.457036	46.187042	51.531296	0.457036	46.186976	51.531296	0.455736	46.198658	51.543675

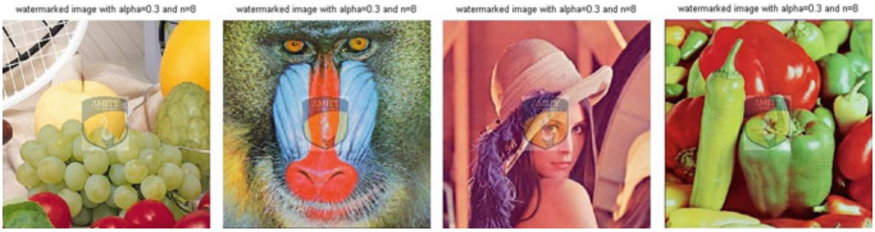

watermarked image with alpha=0.3 and n=8
watermarked image with alpha=0.3 and n=8
watermarked image with alpha=0.3 and n=8
watermarked image with alpha=0.3 and n=8

Fig. 9 Output of watermarking of input image fruits, baboon, lena, peppers watermarking constant as with $\alpha = 0.3$, $\eta = 8$

References

1. Mohanarathinam A, Kamalraj S, Prasanna Venkatesan GKD et al (2019) Digital watermarking techniques for image security: a review. J Ambient Intell Humanized Comput. ISSN 1868-5145. https://doi.org/10.1007/s12652-019-01500-1
2. Hector S, Eduardo F, Rogelio R, Clara C, Mariko N. Visible watermarking technique based on human visual system for single sensor digital cameras. Sec Commun Netw 2017. ISSN-1939-0122. https://doi.org/10.1155/2017/7903198
3. Yongjian H, Kwong S, Huang J (Jan 2006) An algorithm for removable visible watermarking. IEEE Trans Circ Syst Video Technol 16(1):129–133. https://doi.org/10.1109/TCSVT.2005.858742
4. Mohanty SP, Ramakrishnan KR, Kankanhalli MS (July 2000) A DCT domain visible watermarking technique for images. In: Proceeding IEEE international conference multimedia and exposure 2:1029–1032
5. Agarwal N, Singh AK, Singh PK (2019) Survey of robust and imperceptible watermarking. Multimed Tools Appl 78, 8603–8633. https://doi.org/10.1007/s11042-018-7128-5
6. Kim DW, Choi YG, Kim HS, Yoo JS, Choi HJ, Seo YH (2010) The problems in digital watermarking into intra-frames of H.264/AVC. Image Vis Comput 28(8):1220–1228
7. Coatrieux G, Guillou CL, Cauvin JM, Roux C (2009) Reversible watermarking for knowledge digest embedding and reliability control in medical images. IEEE Trans Inf Technol Biomed 13(2):158–165
8. Begum M, Uddin MS (2020) Digital image watermarking techniques: a review. Information 11:110
9. Huang PM, Tsai WH (August 2003) Copyright protection and authentication of grayscale images by removable visible watermarking and invisible signal embedding techniques: a new approach. In: Presented at the conference computer vision, graphics and image processing, Kinmen, Taiwan, ROC
10. Liu T, Tsai W (2010) Generic lossless visible watermarking—a new approach. IEEE Trans Image Process 19(5):1224–1235. https://doi.org/10.1109/TIP.2010.2040757
11. Lu W, Lu H, Chung FL (2006) Robust digital image watermarking based on subsampling. Appl Math Comput 181(2):886–893
12. Lin P-Y, Chen Y-H, Chang C-C, Lee J-S (2013) Contrast-adaptive removable visible watermarking (CARVW) mechanism. Image Vision Comput 31:311–321
13. Luo L, Chen Z, Chen M, Zeng X, Xiong Z (2010) Reversible image watermarking using interpolation technique. IEEE Trans Inf Forensics Secur 5(1):187–193
14. Mir N, Khan MAU (2020) Copyright protection for online text information: using watermarking and cryptography. 2020 3rd International conference on computer applications & information security (ICCAIS), pp 1–4. https://doi.org/10.1109/ICCAIS48893.2020.9096817
15. Sarabia-Lopez J, Nuñez-Ramirez D, Mata-Mendoza D, Fragoso-Navarro E, Cedillo-Hernandez M, Nakano-Miyatake M (2020) Visible-imperceptible image watermarking based on reversible

data hiding with contrast enhancement. 2020 International conference on mechatronics, electronics and automotive engineering (ICMEAE), pp 29–34. https://doi.org/10.1109/ICMEAE51770.2020.00013

16. Perez-Daniel KR, Garcia-Ugalde F, Sanchez V (2020) Watermarking of HDR images in the spatial domain with HVS-imperceptibility. IEEE Access 8:156801–156817. https://doi.org/10.1109/ACCESS.2020.3019517
17. Abraham J, Paul V (Jan 2019) An imperceptible spatial domain color image watermarking scheme. J King Saud Univ—Comput Inf Sci 31(1):125–133
18. Fragoso-Navarro E, Cedillo-Hernández M, Nakano-Miyatake M, Cedillo-Hernández A, Pérez-Meana HM (2018) Visible watermarking assessment metrics based on just noticeable distortion. IEEE Access 6:75767–75788. https://doi.org/10.1109/ACCESS.2018.2883322
19. Chitra K, Venkatesan VP (2016) Spatial domain watermarking technique: an introspective study. In: Proceedings of the international conference on informatics and analytics (ICIA-16). Association for Computing Machinery, New York, NY, USA, Article 50, pp 1–6
20. Barni M, Bartolini F, Piva A (2001) Improved wavelet-based watermarking through pixel-wise masking. IEEE Trans Image Process 10(5):783–791

Survey: Emotion Recognition from Text Using Different Approaches

Aanal Shah, Madhuri Chopade, Parth Patel, and Parin Patel

Abstract Text Processing is a method for comprehending, analyzing, and cleaning text as well as performing actions on the same data. The technique is used to extract meaningful data from text. It is a written form of communication to express emotions through text. Happy, neutral, fear, sadness, surprise, disgust, and anger are the most common emotional expressions. As a result, in the social media era, identifying emotions from text is especially important. A survey of operational methods and approaches for identifying emotion from textual data is discussed in this paper. This research primarily focuses on existing datasets and methodologies that incorporate a Lexical keyword, Machine Learning and Hybrid-based approach.

Keywords Machine learning · Emotion detection · Human-computer interaction · Lexicon-based

1 Introduction

The growing need for emotion recognition from many sources has resulted from advances in computational linguistics and Natural Language Processing (NLP). Emotions can manifest themselves in a variety of ways. The most common and widely used emotion classification is of Ekman's model, in which he classified emotion in six

A. Shah (✉) · M. Chopade
Gandhinagar Institute of Technology, Gandhinagar, India
e-mail: aanalshah2001@gmail.com

M. Chopade
e-mail: madhuri.chopade@git.org.in

P. Patel
Leelaben Dashrathbhai Ramdas Patel (L.D.R.P.) Institute of Technology & Research, Gandhinagar, India
e-mail: patelparth20005@gmail.com

P. Patel
Ahmedabad Institute of Technology, Ahmedabad, India
e-mail: parin.patel2003@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_31

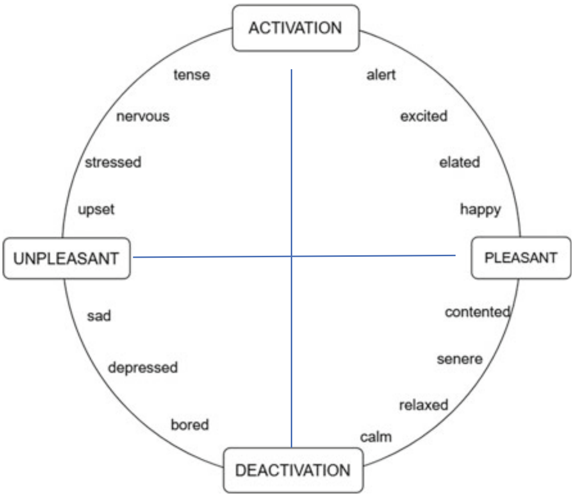
categories (Happy, Anger, Sad, Disgust, Fear, Surprise). After the research many new areas were introduced in affective computing and sentiment analysis [1]. Human-computer interaction (HCI) focuses on the detecting human emotion from nonverbal data. However, still there is lot of confusion and obstacles to get better accuracy over emotion recognition from text [2].

According to psychological research, there are several theories regarding how to express emotions, but two are the most essential and widely employed in existing Sentiment Analysis techniques: emotional categories and emotional dimensions [1]. The category model assumes that emotions are divided into separate groups. This method incorporates Ekman's [1] fundamental emotion model. ANGER, DISGUST, FEAR, HAPPINESS, SADNESS, and SURPRISE were identified as the six primary emotions by Ekman. Plutchik identifies eight fundamental bipolar feelings, which are a superset of Ekman's plus two additional emotions: TRUST and ANTICIPATION. Anger vs. fear, joy versus sadness, surprise versus anticipation and trust versus disgust, are the four bipolar groups into which these eight emotions are classified. Dimensions of emotion affects are represented in a. three-dimensional shape via methods (Fig. 1).

In this model, every emotion has its own proximity [3]. Among all the models, the most distinctive and popular method is of the Rusell's method. Emotions are organized in a two-dimensional circular region, according to Rusell's [4] Circumplex Model of Affect: arousal dimension and valence dimension. UNPLEASANT and PLEASANT feelings can be detected by the valence dimension. The arousal component categorized between DEACTIVATION and ACTIVATION phases (Fig. 2).

In this paper, a systematic survey of the current status, databases and future dimensions. Thus, we have categorized mainly three categories. (1) Rule-based approach (2) Machine Learning-based approach (3) Hybrid approach. In Sect. 5 of this paper, databases, which are necessary to deploy the successful model are discussed and

Fig. 1 Emotion model



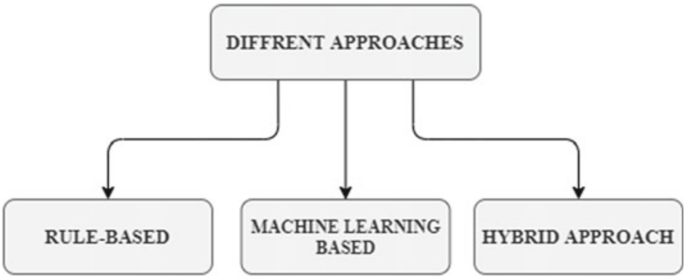


Fig. 2 Different types of approaches

concludes the paper with some analysis, comparison and some points to improve research on this area of affective computing.

2 Rule-Based Approach

Rule-based categorization applies to categories of the emotions in user evaluations through the set of "if–then" rules. The "if" clause is regarded as "rule antecedent", and the "then" clause is known as a "rule consequent" clause. [5]. Rules can be easily created; however, this is a tedious and time-consuming process. Mostly, the rule-based approach consists of keywords recognition and lexicons [6].

2.1 Keyword Based Approach

In this approach the emotion is predicated using the concept of keyword independence, it mostly excludes the possibility of expressing complex emotions using many sorts of keywords at the same time [7]. To check the meaning of the word this model has to deal with numerous dictionaries and lexicons. Such as, WordNet-affect [8]. They devised a simple algorithm for detecting emotive terms in the sentence and computing a result that reflected the word frequency from the text's subjective lexicon [8].

At the beginning of the step, words will get tokenized and prevent the pronouns and prepositions to get enter in the process as they do not directly contribute to the emotion [9]. After finding the lexicon words, a label is given to the sentence by matching the relevant frequency. Among the term complexity and the unavailability of linguistic data, inadequate keywords might have a significant impact on the approach's efficiency [6].

2.2 Lexical Affinity Method

Though the keyword technique is easy and straightforward to implement, it has some barriers which can be resolved by the affinity method. As there are many words that have different usage according to the sentence and emotion associated with them [7].

Aside from emotional keywords, there are a few other things to note. The Lexical Affinity method is a development in the keyword detection approach; it gives a probabilistic “affinity” for a specific emotion to arbitrary phrases [10]. The probability that this technique assigns is part of linguistic corpora. The given possibility is prejudiced toward one particular content type, and it does not identify sentiments derived from the words that do not exist globally on which this methodology relies on, which are some of its flaws [11]. For instance, because the word “accident” has a high likelihood of suggesting a negative feeling, sentences like “I averted an accident” or “I met my lover by accident” would not contribute appropriately to the emotional evaluation.

This lexicon-based approach has mainly 3 resources to develop the relevant meaning of words. (1) utilize the vocabulary of emotions from DUTIR1. (2) Gather a few slang phrases and utilize them. (3) To expand the vocabulary, assemble a collection of emoticons from the short blog website [11]. Two types of lexicon-based methods based on sentiment lexicon include dictionary-based and corpus-based techniques.

In general, a dictionary keeps track of words in a language in a systematic way, whereas a corpus keeps track of text in a language at random [12].

2.3 Statistical Approach

Latent Semantic Analysis (LSA) is the statistical technique to evaluate the links between a group of texts and phrases they include in those documents in addition to creating a set of relevant features in most knowledge-based works [13, 14]. To determine the contextual information between words and sentiment phrases efficiently, the Hyperspace Analogue to Language (HAL) was used. In 2013, wang has suggested a new approach that uses an efficient and better LSA for emotion classification of text using the dataset of ISEAR [15]. This method is also regarded as the lexical based method.

3 Machine Learning Based Approach

To address the problem in a new and effective way, machine learning-based approaches are being used. The challenge with rule-based approaches used to be determining emotions from provided texts, but now the issue is classifying the incoming data/text into various emotions. Machine Learning techniques try to recognize emotions by employing a learned classifier/model that may be applied to a variety of machine learning topics, such as SVM, KNN, naive Bayes (NB), and CRF to determine which emotion category [anger, sadness, joy, fear, disgust, trust, and surprise] should be used.

In these techniques, the emotion is recognized through classification methodologies, which relies on previously trained samples. Hence, the concept of machine learning-based categorization is also known as supervised learning as the model is guided by pre-trained or pre-classified data. Ref. [16] Rather than following solely explicitly coded instructions, such algorithms build a model from different data sources and then utilize that model to make judgements [17].

In emotion detection, categorical procedures are the most common by Calvo [3]. The model by Alm [18] was used to create one of the initial pieces. Roth conducted an empirical study of employing supervised machine learning using the architecture of SNoW learning [4] which has been described in this proposal. They employed a corpus with an expanded collection of Ekman fundamental emotions that were annotated. In one of the experiments given in their paper, Strapparava [8] utilized a Naive Bayes classification algorithm supervised on LiveJournal.com blog postings. They used blog postings that had Ekman's emotions as references (Fig. 3).

We can build emotion-based models using both category and dimensional approaches in studies that use supervised learning algorithms. To detect the writer's emotion class Balabantaray has provided a classification model [19] which is based on SVM multi-class and makes decisions based on Ekman's [1] basic emotions. Roberts [20] uses the emotion of LOVE as one of Ekman's six fundamental emotions. To recognize each of the seven emotions, their method employs a series of binary SVM classifiers to recognize each of the seven emotions. Suttles [21] categorizes emotions based on the 8 fundamental bipolar emotions which were given by Plutchick in previous categorical emotion modelling work. As a result, they might consider emotion recognition to be a discrete problem, apart from a multi-class problem. Strapparava [8] devised a framework, which employs several types of Latent Semantic

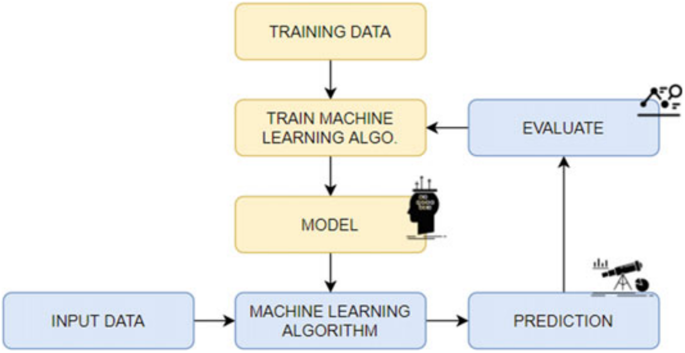


Fig. 3 Machine learning based approach

Analysis to identify emotions when there are no affective words in a text. However, their technique is inaccurate since it is not context-sensitive and fails to capture a conceptual examination of the statement.

Burget [22] proposed a system that significantly relies on pre-processing and labeling the incoming data (Czech Newspaper Headlines) with a classifier. Pre-processing was done at both the levels, sentence and word levels, by employing POS tagging, tokenization, and removing stop-words. To compute the importance between each term and each emotion class (TF-IDF), Inverse Document Frequency was employed. They were able to achieve a prediction performance of 80% on 1000 Czech news heads-6 lines using SVM and tenfold cross-validation. Their approach, however, was not validated on an English dataset. Also, because it solely considers emotional keywords as attributes, it is not context-sensitive.

Dung [23] based his argument on the idea that emotions are connected to cognitive processes that are triggered through emotional events. This indicates that when a specific event occurs, the human mind originates from one mental state and then transforms into another. They used a Hidden Markov Model (HMM) to implement this notion, in which each sentence is made up of several sub-ideas, each of which is regarded as an incident that involves a state change. The algorithm received an Fscore of 35% on the dataset of ISEAR [24], with the greatest precision of 47%. The system's poor accuracy was owing to its failure to evaluate the sentence's semantic and syntactic analysis, making it non-context sensitive.

4 Hybrid Approach

Emotions are detected in hybrid approaches by combining emotion-based keywords and machine learning features obtained from allocated training datasets with knowledge from many areas, such as human psychology [25]. There has been few research on the difficulty of extracting feelings from literature, which neglects keywords based on emotion [25–28].

Wu and others [25] suggested technique for phrase emotion mining is based on identifying preset conceptual tags and sentence characteristics, then classifying just one emotion, joyful, related to biological patterns of human emotions. This was an ambiguous method when one EGR may include multiple emotions.

By establishing common action histogram between two entities, Cheng-Yu and others. Ref. [26] accomplished vent-level textual emotion detection. Each column represents the degree to which the two entities shared an action (verb). They received an F-score of 75% when tested on four emotions. On the other hand, their technique ignores the content of the phrase and relies largely on the structure of the training data, such as the grammatical type of sentences in the data and the frequency of emotions for a certain subject. Furthermore, only four of the six Ekman emotions are used in the categorization.

Chaumartin [27] created UPAR7, a framework based on linguistic rules based on the lexical resources], SentiWordNet [29], WordNet-Affect [17] and WordNet

[30]. This is based on the Stanford POS tagger's dependency graph [31], which is used by the system, with the root nodes of the derived graph serving as the main subject. For each emotion, each word in the statement is assessed separately. Because the principal objective (major word) is more important than the other words in the sentence, it receives a higher grade. This method's best accuracy was 30% for the Ekman model's six emotions. This method is not context-sensitive and lacks a global comprehension of the language, in addition to its low accuracy.

For text-based emotion recognition, BERT is the most studied transformer-based model. The research suggests that these BERT variants be investigated in terms of detecting emotions in textual data [13]. The LSTM Bi-directional words embedding and annotated corpus were proposed by [32]. The first stage is to apply a preprocessing technique to the input data, in which we remove excessive spaces, incorrect characters, resolve character encoding, and do spelling correction.

Yang [33] suggested a machine-learning-based emotion classification system that combines CRF-based (conditional random field) emotion trigger identification, phrase-based detection, and SVM, Naive Bayesian, and Max Entropy-based emotion classification. The system performed well on a dataset of suicide notes, with an Fscore of 61%, precision of 58%, and recall of 64%. This strategy produced reasonable results, but neither the classifier nor the dataset is published.

5 Dataset

The gathering of data relevant to the course is the next essential stage in recognizing emotions from the text after settling on the model to represent emotions. For research purposes, there are a few structured annotated datasets for emotion detection that are freely available. This section lists the most important publicly accessible datasets and their characteristics. Table 1 lists the datasets, their characteristics, and the emotion models they reflect.

Table 1 Characteristics of dataset

Dataset	Emotion models	References
ISER	Discrete	[34]
EMOBANK	Categorical and dimensional	[35]
SemEval	Discrete	[36]
Emolnt	Discrete	[37]
Emotion-Stimulus	Discrete	[38]

Table 2 Characteristics of ISEAR database

Emotions	No. of examples
Anger	1096
Disgust	1096
Fear	1095
Sadness	1096
Shame	1096
Joy	1094
Guilt	1093
Total Examples	7666

5.1 ISEAR Dataset

The world-level study on Emotion Antecedents and Reactions (ISEAR) project, headed by Harald Wallbott and Klaus R. Scherer, gathered data from a large group of psychologists around the globe in which 7 emotion labels were declared (joy, sadness, fear, anger, guilt, disgust, and shame). It was obtained through the questionnaire from 37 different countries and 3000 respondents from 5 continents [34] (Table 2).

5.2 Emobank

EmoBank is a collection of over 10,000 phrase corpus labelled with multidimensional emotional metadata in the Valence-Arousal-Dominance (VAD) style, combining various genres. EmoBank is not only bi-representational but also bi-perspectival in design, which makes it stand out [35]. It is consisting of the reader's and writer's emotions. The automatic mapping among categorical and dimensional makes the dataset feasible and efficient to use in machine learning techniques.

5.3 SemEval-2017 Task 4

The Semantic Evaluations (SemEval) database contains Arabic and English news headlines from different sources. The task adds substantial benefits to the sentiment community by making a large, publicly accessible benchmark dataset with over 70,000 tweets in two languages available for academics to examine and compare their approach to the current [36].

5.4 WASSA-2017 Emotion Intensities (EmoInt)

To estimate emotion intensities in tweets, researchers analyzed data from the seminar on Quantitative Methodologies to social media, Sentiment and Subjectivity. For the four feelings (fear, pleasure, rage, and sorrow), training and testing data were represented. For instance, the predicted value of the user's anger was measured in 0 and 1, which describe the level of anger along with that, it has a corresponding tweet regarding that anger text [37].

5.5 Emotion-Stimulus Dataset

The Emotion-Stimuli Dataset was created by Ghazi et al. in 2015 and using FrameNet's emotions-directed frame, both the emotion and the stimuli were validated. There are categories for happiness, sorrow, anger, fear, surprise, disgust, and humiliation. There are 820 sentences having both a cause and an emotion tag, and 1594 sentences with only an emotion tag and has 2414 items in XML file. This dataset was built carefully and in well-manner form.

6 Comparison

All of these approaches are appropriated in certain manners. Most experiments and papers are demonstrated that comparatively hybrid methods perform well than learning and rule-based performs solely. Ahead of these approaches Poonam [34] suggested their method, which seems to perform better than previous ideologies. Their results are depicted below, each row retrieved from their results (Table 3).

Based on the research findings of the experiment, a graphical comparison of all significant approaches to emotion recognition is shown below (Fig. 4).

Table 3 Comparison of different approaches

	Keyword (%)	ML-based (%)	Hybrid (%)	Poonam [34] method (%)
Precision	59.98	64.89	75.90	75.16
F-measure	61.29	65.42	76.15	76.98
Recall	62.66	65.97	68.39	76.06
Accuracy	57.50	63.68	65.23	81.89

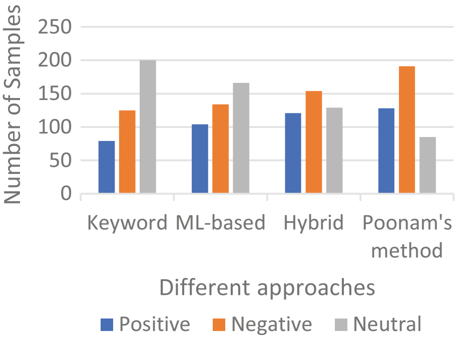


Fig. 4 Comparison of different approaches for detecting emotions

7 Limitations and Future Scope

Dimensional and discrete models are used in the current TER tasks. However, universal annotation standards should be established, as their lack shows incompatibilities among existing techniques. SemEval [36] is annotated with Ekman's six basic emotions, for example, Wassa-2017 (EmoInt) Fear, pleasure, rage, and sorrow are the four emotions annotated in the dataset [37]. Various emotion labels have been proposed, each of which has an impact on the compatibility of cross-corpus resources. This limitation (incompatible notation) can be used to indicate a lack of data for training and testing [14] (Table 4).

In future work, solving this limitation of incompatibility between model for labelling various emotions will enhance integration of existing corpus resources. We highly encourage future researchers to design substantial large-scale dataset which includes micro-scale emotion too, which can be helpful to train model effectively. Along with this, creating efficient generative deep learning model will be more effective solutions with large dataset.

Table 4 Limitation of current works in text-based emotion detection

Proposed work	Approach	Dataset	Limitation
Seal et al. [9]	Rule based	ISEAR data	Poor words vocabulary in the lexicon
Ahmad et al. [39]	Machine Learning	Emo-Dis-HI data	Neglecting the contextual meaning of words
Matla and Badugu [28]	Machine Learning	Tweets	Contextual information in sentences is extracted insufficiently
Hasan et al. [40]	Hybrid	Tweets	Weak semantic feature extraction

8 Conclusion

In this survey, we discussed mainly three approaches to classify or recognize the emotion based on text data on the fine-grade level. Our contribution readdresses the techniques and concluded that among the presented approaches hybrid method seems logical when the community or team have enough trained data as well as a strong keyword for employing the rule-based methodology. On the other hand, in the particular learning approaches, clearly observed by several researchers that Support-vector-machine (SVM), Naïve Bayesian, and KNN derived impressive outcomes. It is also noticeable overall that the classifier model is context-sensitive thanks to syntactic and semantic analysis of the sentence, and the use of ConceptNet and Wordent assist the algorithm in characterizing the training dataset, resulting in improved coverage of emotion rules. We have also taken various wide-range datasets into consideration, which can be helpful in the case trained own model or performing analysis over the data. The entire survey has identified that, rather than giving an individual emotional rate to each word, examining the relationships between the terms of the sentence could lead to greater accuracy.

References

1. Paul E (1999) Basic emotions. In handbook of cognition and emotion, pp 45–60; Francisco V, Gervás P (2013) EmoTag: an approach to automated mark-up of emotions in texts. Comput Intell 29(4):680–721.
2. Sebe N, Cohen I, Gevers T, Huang TS (2005)Multimodal approaches for emotion recognition: a survey. Proceedings of SPIE—the international society for optical engineering, vol 5670, 08, pp 56–67. https://doi.org/10.1117/12.600746
3. Calvo RA, Kim SM (2013) Emotions in text: dimensional and categorical models. Comput Intell 29(3)
4. Roth D, Cumby C, Carlson A, Rosen J (1999) The SNoW learning architecture. Technical report, UIUC Computer Science Department; Russell JA (1980) A circumplex model of affect. J Pers Soc Psychol 39(6):1161–1178
5. Asghar MZ, Khan A, Ahmad S, Qasim M, Khan IA (2017) Lexicon-enhanced sentiment analysis framework using rule-based classification scheme. PLoS ONE 12(2):e0171649. https://doi.org/10.1371/journal.pone.0171649
6. Acheampong FA, Wenyu C, Nunoo-Mensah H (2020) Text-based emotion detection: advances, challenges, and opportunities. Eng Rep 2:e12189. https://doi.org/10.1002/eng2.12189
7. Kao E, Liu CC, Yang T-H, Hsieh C-T, Soo V-W (2009) Towards text-based emotion detection: a survey and possible improvements. Proceedings—2009 international conference on information management and engineering, ICIME 2009, pp 70–74. https://doi.org/10.1109/ICIME.2009.113
8. Strapparava C, Mihalcea R (2008) Learning to identify emotions in text. In: Proceedings of the ACM symposium on applied computing, pp 1556–1560. https://doi.org/10.1145/1363686.1364052
9. Seal D, Roy UK, Basak R (2020) Sentence-level emotion detection from text based on semantic rules. In: Tuba M, Akashe S, Joshi A (eds) Information and communication technology for sustainable development. Advances in intelligent systems and computing, vol 933. Springer, Singapore. https://doi.org/10.1007/978-981-13-7166-0_42

10. Shivhare SN, Khethawat S (2012) Emotion detection from text. Comput Sci Inf Technol 2. https://doi.org/10.5121/csit.2012.2237
11. Chopade R (June 2015) Text based emotion recognition: a survey. Int J Sci Res (IJSR) 4(6):409–414. https://www.ijsr.net/search_index_results_paperid.php?id=SUB155271
12. Nandwani P, Verma R (2021) A review on sentiment analysis and emotion detection from text. Soc Netw Anal Min 11:81. https://doi.org/10.1007/s13278-021-00776-6
13. Acheampong FA, Nunoo-Mensah H, Chen W (2021) Transformer models for text-based emotion detection: a review of BERT-based approaches. Artif Intell Rev 54:5789–5829. https://doi.org/10.1007/s10462-021-09958-2
14. Deng J, Ren F. A survey of textual emotion recognition and its challenges. In: IEEE transactions on affective computing. https://doi.org/10.1109/TAFFC.2021.3053275
15. Wang X, Zheng Q (2013) Text emotion classification research based on improved latent semantic analysis algorithm. https://doi.org/10.2991/iccsee.2013.55
16. Acheampong FA, Wenyu C, Nunoo-Mensah H (28 May 2020) Text-based emotion detection: advances, challenges, and opportunities. Wiley Online Library
17. Bishop CM (2006) Pattern recognition and machine learning. Springer; Bradley MM, Lang PJ (1999) Affective norms for English words (ANEW): instruction manual and affective ratings. Technical report, The Center for Research in Psychophysiology, University of Florida
18. Alm CO, Roth D, Sproat R (2005) Emotions from text: machine learning for text-based emotion prediction. In: Processing conference human language technology and empirical methods in natural language processing, pp 579–586
19. Balabantaray RC, Mohammad M, Sharma N (2012) Multi-class Twitter emotion classification: a new approach. Int J Appl Inf Syst (IJAIS) 4(1):48–53
20. Roberts K, Roach MA, Johnson J, Guthrie J, Harabagiu SM (2012) EmpaTweet: annotating and detecting emotions on Twitter. In: Calzolari N (Conference Chair) Piperidis, Choukri K, Declerck T, Doğan MU, Maegaard B, Mariani J, Moreno A, Odijk J, Stelios (eds) Proceedings of the eight international conference on language resources and evaluation (LREC' 12). European Language Resources Association (ELRA)
21. Suttles J, Ide N (2013) Distant supervision for emotion classification with discrete binary values. In: Gelbukh A (ed) Computational Linguistics and intelligent text processing, volume7817 of lecture notes in computer science. Springer, Berlin Heidelberg, pp 121–136
22. Burget R, Karasek J, Smekal Z (2011) Recognition of emotions in Czech newspaper headlines. Radioengineering 20(1):39–47
23. Ho DT, Cao TH (2012) A high-order hidden markov model for emotion detection from textual data. In: Richards D, Kang BH (eds) Knowledge management and acquisition for intelligent systems. PKAW 2012. Lecture notes in computer science, vol 7457. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-32541-0_8
24. Scherer KR, Wallbott HG (Feb 1994) Evidence for universality and cultural variation of differential emotion response patterning. J Pers Soc Psychol 66(2):310–28. https://doi.org/10.1037//0022-3514.66.2.310; (Jul 1994) Erratum in: J Pers Soc Psychol 67(1):55. PMID: 8195988
25. Chung-Hsien W, Chuang Z-J, Lin Y-C (2006) Emotion recognition from text using semantic labels and separable mixture models. ACM Trans Asian Lang Inf Process (TALIP) 5(2):165–183
26. Cheng-Yu L et al (2010) Automatic event-level textual emotion sensing using mutual action histogram between entities. Expert Syst Appl 37(2):1643–1653
27. Chaumartin F-R (2007) UPAR7: a knowledge-based system for headline sentiment tagging. Proceedings of the 4th international workshop on semantic evaluations. Association for computational Linguistics
28. Suhasini M, Srinivasu B (2020) Emotion detection framework for Twitter data using supervised classifiers. New York, NY, Springer, pp 565–576
29. Esuli A, Sebastiani F (2006) Sentiwordnet: a publicly available lexical resource for opinion mining. Proceedings of LREC, vol 6
30. Miller GA (1995) WordNet: a lexical database for English. Commun ACM 38(11):39–41

31. Toutanova K, Klein D, Manning C, Singer Y (2003) StanfordPOStagger, [Online]. Available: http://nlp.stanford.edu/software/tagger.shtml,Stanford
32. Rashid U, Iqbal MW, Skiandar MA, Raiz MQ, Naqvi MR, Shahzad SK (2020) Emotion detection of contextual text using deep learning. 2020 4th International symposium on multidisciplinary studies and innovative technologies (ISMSIT), pp 1–5. https://doi.org/10.1109/ISMSIT50672.2020.9255279
33. Yang H et al (2012) A hybrid model for automatic emotion recognition in suicide notes. Biomed Inf Insights 5(Suppl 1):17
34. Arya P, Jain S (May–June 2018) Text-based emotion detection. IJCET 9(9)
35. Scherer KR, Wallbott HG (1994) Evidence for universality and cultural variation of differential emotion response patterning. J Pers Soc Psychol 66(2):310
36. Buechel S, Hahn U (2017) Readers versus: writers versus texts: coping with different perspectives of text understanding in emotion annotation. Paper presented at: proceedings of the proceedings of the 11th Linguistic annotation workshop, pp 1–12
37. Rosenthal S, Farra N, Nakov P (2019) SemEval-2017 task 4: sentiment analysis in Twitter. arXiv preprint arXiv:1912.00741
38. Mohammad SM, Bravo-Marquez F (2017) WASSA-2017 shared task on emotion intensity. arXiv preprint arXiv:1708.03700
39. Ahmad Z, Jindal R, Ekbal A, Bhattachharyya P (2020) Borrow from rich cousin: transfer learning for emotion detection using cross lingual embedding. Expert Syst Appl 139:112851
40. Huang C, Trabelsi A, Zaïane OR (2019) ANA at SemEval-2019 Task 3: contextual emotion detection in conversations through hierarchical LSTMs and BERT. arXiv preprint arXiv:1904.00132

A Survey on Vision-Based Elders Fall Detection Using Deep Learning Models

Shital N. Patel, Amit Lathigara, Viswash Y. Mehta, and Yogesh Kumar

Abstract Falls are a leading source of severe wounds in adults over the age of 60, according to WHO. It has the potential to cause significant wounds, and impairments, and can even lead to demise, particularly in older citizens who live alone. As a result, we can improve these people's quality of life by implementing different automated fall detection systems. Using multiple deep learning architectural models, this study provides a full-scale evaluation of recent fall detection approaches. Its goal is to serve as a resource for academics and industry to research various fall detection methods. We investigated nearly all contemporary along with promising deep learning approaches for fall tracking and detection and divided them into two classes: convolutional neural network-based (CNN) systems and recurrent neural network-based (RNN) systems. The focus of this research is to offer a contrast of several deep learning-based fall detection systems like CNN, LSTM, and RNN as well as illustrated a comprehensive table that gives an overview of the paper published from the year 2017 to 2021 that could be used in better understanding the role and significance of the systems in the fall detection task. At the closure of the report, obstacles and further development have been mentioned. This survey can assist scholars in better understanding of present systems and proposing new techniques by addressing the challenges that have been identified.

S. N. Patel (✉) · A. Lathigara
School of Engineering, RK University, Rojkot, India
e-mail: patel.sheetalv@gmail.com

A. Lathigara
e-mail: amit.lathigara@rku.ac.in

S. N. Patel
Gandhinagar Institute of Technology, Kalol, India

V. Y. Mehta
Computer Engineering, Ahmedabad Institute of Technology, Ahmedabad, India

Y. Kumar
Mechanical Engineering, Pandit Deendayal Energy University, Gandhinagar, India

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_32

Keywords Fall detection · Deep learning · Convolutional neural network · Recurrent neural network · Review

1 Introduction

Falls are a prominent origin of accidentally wounding among the elderly [1]. Almost 8% to 35% of people aged 65 and over fall each year, increasing up to 32–42% for those over 70 years of age. The fluctuation of collapse escalates with age and deficit of strength amount. Senior people who are living alone fall more often than those who are living in the colony. Roughly, 30–50% of people living in long-term care facilities fall each year, and 40% of them encounter recurrent falls [2]. The estimation of warded admission due to falls for people aged 60 and older in Australia, Canada, and the United Kingdom (UK) ranges from 1.6 to 3.0 per 10,000 population. Fall injury rates consequent in emergency province visits of the same age group in Western Australia and the United Kingdom are greater: 5.5–8.9 per 10,000 population total [3]. Tumbles and the accidents they cause are major public health concerns that frequently need medical attention. Falls are the cause of 20–30% of moderate to traumatic complications and 10–15% of emergency room visits. Over half of all injury-related hospital treatment is provided to those aged 65 and up [2]. Hip fractures, traumatic brain injuries, and upper limb injuries are the most common root factors of fall-related hospitalization [3]. The amount of people over 60 is expanding faster than every age category throughout the globe. In 2006, the number of people in this age group was expected to be 688 million, with an expected increase to over two billion by 2050 [4]. For the first period in world history, the number of elderly people will be substantially bigger than that of youngsters under the age of 14. Furthermore, the eldest sector of the population, those aged 80 and up, is most vulnerable to falls, and the repercussions are the fastest rising within the aging population, estimated to account for 20% of the total by 2050 [4]. It has been tested that approximately 50% of the elderly human's mendacity at the ground due to fall activities for multiple hours extinguishes inside half a year although they now no longer have any straightforward wounds [5].

Over the decades, a lot of effort has been taken for fall detection to grow the accuracy and lessen the fake alarm. By virtue of unbiased living, in absence of an automatic fall recognition system, the emergency providers cannot attain the destination immediately. This ends in excessive issues for a targeted group of the community. It could be very tough to make complete surroundings fall evidence because of negative overall performance and fake alarm. So, there may be a need to broaden an automatic fall detection system that gives excessive overall performance and almost zero fake alarm rate. Here, we have reviewed overall 35 papers which give diverse strategies to come across falls with the aid of using deep learning. Out of 35 papers, a majority of papers make a speciality of CNN primarily based architecture, and the rest are focused on RNN-based architecture. Also, in a few reviewed papers, a mixture of CNN and RNN versions was used which reduces the

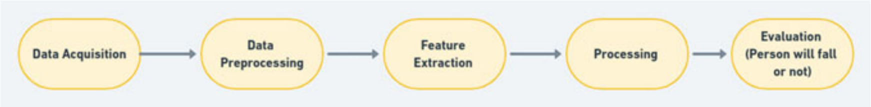


Fig. 1 General model of the fall detection system

trouble of accuracy, key points of features, temporal and spatial information, data occlusion, and foreground object detection.

Deep learning is a subspace of machine learning which turns out to be a subset of artificial intelligence that grants us to attain certain types of tasks and decision automation for numerous domains. Fall detection methods that use deep learning may identify falls by relying on an object's activity patterns. The possible consequences of deep learning algorithms for binary classification in the domain of fall detection might befall or not fall. The target dataset is cleaned and pre-processed to ensure that the end results variables anticipated by deep neural networks are correct. Following that, feature extraction is used to obtain the greatest possible collection of features that will help us train a model more effectively. Figure 1 depicts a broad recapitulation of the fall detection technique.

We present a comprehensive overview of fall detection systems in this survey study, which is intended for wide readers to become familiar with the literature in this area. The chunks of the paper are organized as follows. In Sect. 2, we start with a comprehensive overview of major-related works in this area followed by Sect. 3, which introduces various deep learning approaches such as fall tracking and detection and divided them into two classes: convolutional neural network-based (CNN) systems and recurrent neural network (RNN) that includes the RNN and LSTM-based systems, giving a comprehensive table that gives the overview of the paper published from the year 2017–2021 that could be used in better understanding the role and significance of the systems in the fall detection task. Section 5 includes the challenges and future work, where it illustrates the major challenges in fall detection systems due to lack of data and also points out comparisons between different models to strengthen the accuracy of the fall detection models. The future work includes the development that could happen shortly, giving aspects on applications and techniques that could be used to significantly boost the overall performance of the model. Finally, we provide a conclusion of the survey in Sect. 5.

2 Related Works

Waheed et al. [6] have presented a noise-tolerant deep learning-based system that particularly uses RNN with an underlying Bi-LSTM stack which gathers information from a smartphone, wearable sensors, and multimodal sensors. The recommended approach obtains an accuracy of 97.41 percent and 97.21 percent, respectively, on 2 publically available datasets, Sisfall, and UP-fall detection. However, adopting

CNN, which automatically identifies characteristics from complicated sensor data, can enhance accuracy even further.

Arifa et al. [7] present a design for the classification of fall activities from additional indoor events. They have executed a position distinction technique together with additional activities for altering crucial and instructive positions from the record. They then utilized a two-dimensional CNN architecture to retrieve significant information from video frames and GRU to determine the temporal reliance on human movements. They used a binary cross-entropy loss function to estimate network parameters like weights and training rate to lower the losses. To reduce the losses, they employed a binary cross-entropy loss function to determine network properties like weights and learning rate. Eventually, falls are detected using the sigmoid activation function. They have utilized two common datasets like URFall and multiple camera fall datasets to analyze the model's performance in terms of accuracy and demonstrated that the suggested technique outperformed another existing method. They have not, however, created their own dataset and do not consider outside surroundings while detecting falls, which is a difficult task.

Author in [8] has proposed an optimized Alexnet Convent architecture with data collected from various wearable sensor devices like magnetometers, gyroscopes, and accelerometers. They employed multilinear PCA to determine the dimensions of retrieved data and then used an intelligent Alexnet convolutional network to predict falling with a 99.45% accuracy. They haven't yet made any comparisons between CNN and Bi-LSTM.

The author in [9] has proposed a dynamic video classification system in order to detect falls. They have used Mask R-CNN to extract human body silhouettes from video frames and then used a combination of CNN and LSTM to detect falls between successive frames. They applied the proposed method on the UP-fall dataset and received an accuracy of 100% with a loss of 2% only. However, they have managed to perform experiments in a single building only which limits the supervision activities of all elders at the regional level.

Reference [10] uses impulse-radio ultra-wideband (IR-UWB) monocratic radar to solve the fall detection problem without the need for any devices. To extract spatiotemporal characteristics from the radar range flow of data, the researchers changed the fall detection issue into a spatiotemporal sequence classification problem and suggested a learning model constructed by merging CNN and 1-dimensional convolutional long short-term memory (ConvLSTM). Not only did this suggested technique outperforms the CNN-based approach in regards to 93 percent accuracy and 92.6 percent specificity for fall detection difficulties, but also detected falls with a high sensitivity of 96.8% in richly furnished surroundings. However, the existing system only supports the results in the private room instead of in external environments.

In [11], the sleep state, an energy-efficient wearable sensor was created to sense and store information about various actions, and an impede mechanism was given to transport the information to a ZigBee-connected server. To discriminate between falls and regular living activities, a fall detection deep neural network (FD-DNN) that combines CNN and LSTM was created and deployed on a server. This deep neural network, when evaluated against online and offline datasets, achieved a higher fall detection accuracy score of 99.17%, while its specificity and sensitivity are 99.94% and 94.09% than the traditional classification algorithms. However, they have used too little elderly data to train the model which results in low accuracy.

Yao et al. [12] state that head movement is very important during the fall process due to the high amplitude of head motion. As a result, they devised the forward segmentation approach, which uses two separate ellipses to suit the person's upper torso. They employed a fun Cnn architecture to detect indoor falls and a Gaussian mixture model to captivate the human targeted. The paper concludes that using a combination of GMM and CNN to detect falls results in a 90.5 percent detection rate and just a 10% false alarm rate. They do not, however, give techniques to deal with occlusion in a more realistic interior setting, nor do they investigate the various applications of this strategy outside.

The author in [13] has obtained valued features and the position details by using the You Only Look Once (Yolo) object detection model and real-time human pose detection model (Open Pose) for preprocessing to obtain key points and the position information of a human body. To acquire the ongoing features of the person, a double-track sliding window architecture is delineated. Then, later used multi-layer perceptron (MLP) and random forest to classify the static and dynamic data separately. The paper has achieved 97.33% once validated with the UR FD dataset and attained 96.91% for the Le2i fall detection dataset. However, they have not compared the proposed method with other models like CNN and Bi-LSTM which can give good results and also do not focus on some perilous deportment that may arise in the liveliness of the aged.

In the paper [14], the authors have proposed a CNN along with a long short-term memory network (LSTM) which detects falls efficiently and alerts somebody as quickly as possible they conducted trials with ten individuals in three different places in a 40-square-meter flat. They have demonstrated that UWD radars are a promising new technique for tracking and action recognition, with worldwide accuracy of up to 90%. However, they have not done many efforts to increase the fall detection rate and can achieve a good generalized model by recruiting more participants in the building.

The statement of belief in the paper [15] presents a multi-source CNN ensemble (MCNNE) structure that inputs data from three CNNs: a sensor array, a velocity detector, and a rotational motion detector. The feature data are generated by the fully connected layer, whose dimensions are further reduced by principal component analysis and subsequently by the softmax function, yielding a classification rate for fall detection. The paper's major purpose is to classify the original information into nine categories based on a set of criteria, forward, reverse, rightward, rightward, lying on a bed, bending, running, and lying on the floor. The total average falls recognition rate is close to 99.30 percent, with a false-positive rate of less than 0.69 percent, demonstrating the benefit of an ensemble method. Furthermore, expanding the number of model inputs and the accompanying data amount can help the model generalize even more.

Cai et al. [16] A multi-task hour glass convolutional auto-encoder (HCAE) was used to overcome the problem of information loss after numerous layers of a deep learning model in a vision-based fall detection technique. Hourglass residual units (HRUs) are added into the encoder to allow neurons with contextual areas to acquire multiscale characteristics such as faces and hands. The intermediate feature will then be given into a classification to assess falls, and the decoding will finally be utilized to recreate the original frames. The experimental results show that the proposed method achieved an accuracy of 96.2% with the sensitivity, specificity, precision, and F-score are 100%, 93%, 92.3% and 96%, respectively. However, they have not done any experiments on some complicated environments which can further improve the lives of the elderly.

In the paper [17], the author demonstrated a deep learning strategy that ascertains falls of the elderly by exploiting an unsupervised model known as spatiotemporal residual auto-encoder (SRAE). By use of ConvLSTM, the model is able to extract nonlinear and secular features of infrared thermography (IRT) videos. Fall actions, as well as normal ADL, are being tested on the proposed model in the testing phase. The result confirmed that the suggested model had higher accuracy and even better performance than the other models shown. They can also apply the proposed method for depth camera-based datasets to more assess the model's generalization capacity.

In [18] the fall detection dataset (FDD), URFD and the MCF dataset come to taken into consideration for the experimentation of the VGG-16 convolutional neural net model due to its high accuracy and blend reciprocal details such as oriented optical flow, shape, red green blue, and amplitude to intensity to produce a more precise identification than if only a sensory was used. To learn domain-dependent features and to increase the number of images in fall datasets, techniques like data augmentation and transfer learning are used. The paper achieved an accuracy of 99.72% with data augmentation whereas 96% accuracy without data augmentation. They also need to include sensor-based data and audio information to further increase the accuracy.

The main aim of the paper [19] is to detect fall events in complex backgrounds based on visual data. To precisely obtain the motion entities in the noise background, the Mask R-CNN method was acquired. For eventual fall event detection, attention is driven bi-directional LSTM (long short-term memory) architecture was used. The paper pointed out that the Mask R-CNN with attention guided bi-directional LSTM model executed superior in the self-augmented dataset. However, they have only done experiments on a single person which cannot further improve the lives of the elderly.

In [20], multimodal fall detection techniques have been developed to decrease false alarms in tracking. They used CNN and LSTM for real-time identification on a multimodal dataset dubbed UP-fall detection, which included 12 activities done by 17 people. They have compared their results with baseline RF, CNN, and LSTM and proved that CNN and LSTM combination can reach up to the accuracy of 96.4% which is high compared to existing methods. However, they can further improve the generalization capability of the model by testing other aggregation and fusion-based methods also.

In paper [21], they have proposed an image-based fall detection strategy to extract motion features through an optical flow method and used a convolutional neural network (CNN) to extract the input features. Additionally, to improve the performance of the system, a dataset for fall detection comprises video recordings of one subject performing different types of falls. Finally, by applying the testing set, the fall detection system performed with 92.78% of accuracy and other metrics can also be computed, such as precision (95.27%), recall (95.42%), and F1-score (95.34%). The output of 96.84% accuracy was achieved after accomplishing the voting strategy which improved the overall performance of the model. The performance of the model can be further improved using real-time scenarios and also include motion history in fall activity.

Reference [22] The author in order to tackle the difficulty of recognizing falls, has proposed a pre-trained CNN Alexnet which takes continuous wavelet transformation to get to obtain as input a time–frequency characterization of the bio-radar signals. Here, CNN1 and CNN2 are trained on bio-radar 1 and bio-radar 2 individually, and CNN12 is trained on both bio-radar 1 and bio-radar 12. The result shows that CNN12 generates an accuracy of 99.29% which is high compared to CNN1 and CNN2. Additionally, sensitivity is 98.57%; specificity is 100%; precision is 100%, and F1_score is 99.28% which is also high compared to CNN1 and CNN2. They could, therefore, expand the study by accounting for variations in ambient circumstances and the presence of various forms of deformations.

The CNN-based model labeled CNN-3b3Conv is less complex compared to CNN is proposed [23] and comprises 3 convolution layers, 2 max pooling, and 3 fully connected layers. They have established an Internet of Things permitted fall identification system using fog computing and deep learning. Here, three open datasets—the URFD dataset, smart watch dataset, and notch dataset—are used to evaluate the performance of CNN-3B3Conv. They have compared evaluated results with LSTM Acc, LSTM Acc Rot, and CNN-3BN3Conv absence of data amplification and proved that CNN-3BN3Conv with data augmentation gives an accuracy of 99.86% which is high compared to others. They can increase the performance of the model further by improving the algorithm's capability to handle real-time events and by focusing on more fall specific features.

In [24], they demonstrated multi-camera fall recognition and tracking using a vision system that utilizes Cnn architectures with visual information derived from picture sequences using the optical flow approach. Researchers conducted a study employing optical flow-based characteristics from both cameras (Cam1 and Cam2) simultaneously and found that integrating a side and front image of camera1 and camera2 yields an accuracy of 95.64 percent, which is higher than a single camera. The findings show that our suggested fall detection system has a prediction power of 97.43%, outperforming traditional ML approaches by exploiting optical flow-based characteristics.

The authors in [25] proclaimed it to be the first neural network explanation for fall detection without self-made feature extraction. To abstract both spatial and temporal attributes of the video pattern which has well encoded the motion information within the task, they developed a new system by a combinable measure of 3D CNN and

LSTM-based attention mechanism. The current largest video classification benchmark dataset of sports videos Sports-1 M was selected to train the 3D CNN. LSTM is trained on divergent data for particular execution, and the 3D CNN is pre-trained on dataset Sport-1 M which in turn gives pretty adequate results. They can further improve the performance by integrating Bi-LSTM with CNN instead of LSTM.

In [26], the research developed an integral equations, confidentiality, and fast fall detection system using depth photographs utilizing a Kinect Red Green Blue-D sensor. The model is created utilizing a CNN and RNN-based end-to-end pipeline to identify fall actions. The deep ConvNet extracts visual features from the input sequence images, while LSTM recurrent neural networks identify fall events. The ConvLSTM architecture, which blends Convent and LSTM, is built to focus solely on human body motions while ignoring static objects. The suggested approach was tested using public URFD fall detection data and found to be 98 percent accurate. They haven't yet, though, conducted any research in a dark atmosphere at night or in outdoor systems to enhance the lives of the elderly.

Tsai et al. [27] proposed a skeleton information extraction algorithm, which transforms depth information into skeleton information and extracts the critical joints associated with fall activity. After extracting multiple features by connecting multiple frames, they carried out a CNN to understand the human fall. For testing and training purposes, they have used NTU RGB + D, a well-known benchmark in the posture recognition technique and achieved an accuracy of 99.2%. However, the proposed system only focuses on fall activity as opposed to other posture recognition. Additionally, they can enhance the hardware cost by changing the Kinect camera into a normal camera also. The authors in [28] employed the use of a CNN-based method that detects falls. From a video stream, it extracts the human silhouette and detects various posture classes and a fall is detected, while the lying posture is detected at the ground region. Experiments indicate that the proposed CNN classifier outperforms conventional ones in posture classification accuracy of 94.75% for 10 postures. [29] proposes a novel video-based fall detection system using human pose estimators and stereo depth data. A generic 2D human pose estimator is utilized in combination with depth data to estimate 3D human key points and to calculate the ground plane in 3D and classify if a person has fallen or not. The model achieved an accuracy of above 91% while tested against two large test scenarios. However, they have not tested external and complex environments which limits the generalization functionality of the system.

3 Deep Neural Network Models

3.1 Convolution Neural Network (CNN)

CNN is a dense and FFN net framework that is commonly used for evaluating images. [30] The CNN framework's most significant attribute is that it can optimize the CNN's

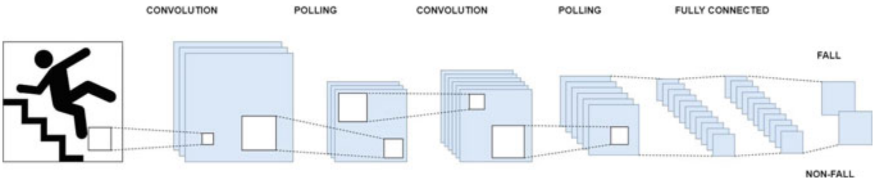


Fig. 2 Convolutional neural network

weight utilizing a large amount of training data without needing time-consuming human effort, leading to reliable classification and identification. The convolutional layers generate a collection of extracted features dependent on the number of selected filters (or weights). These features are used to apply convolution to the input data. The feature convolves across a tiny area of the input and produces an output, which will then be carried along to the next layer as a moving window. A nonlinear activation function such as ReLU, sigmoid, or tanh is utilized after the feature maps have been computed. Different layers are connected to these networks to create appropriate structures. Similar to these layers, a convolutional layer attempts to generate feature representations of the input [30] (Fig. 2).

The following is a basic description of CNN's primary components:

- Convolutional layer—to get numerous mapping feature maps, convolute the input raw picture with several trainable filters (also known as the convolutional kernel) and additive bias vectors.
- Pooling layer—this layer is used for down-sampling behind the convolution operation to minimize the dimensionality of the feature. Max and mean pooling are the two utmost used pooling algorithms.
- Fully connected layer—the generated characteristics are condensed into a one-dimensional vector and used for classification after many convolutional and pooling layers have analyzed the raw image. In this layer, you may add further features to this one-dimensional vector.

Fall recognition approaches that use convolution neural models may leverage CNN's ability to classify photographs since CNN excels at identifying structure and patterns in images. LSTMs can effectively and quickly distinguish between falling and ADL incidents since sensor data on falls are time-specific. Some of the fall detection investigated use a mix of LSTM and CNN to deal with general vision-related difficulties such as image noise, transparency, wrong segmentation, perspective, and so on. Unsupervised learning of effective data coding is done using auto-encoders, while supervised learning is done with CNN and LSTM structures.

For the most part, the investigated strategies use CNN to build their autonomous fall detection systems. Each input picture is sent through a succession of convolutional layers for training and validation, including filters, pooling layers, fully connected layers (FC), and activation functions.

Other variants/successors of CNN are as below:

Deep CNN is often used for object recognition, image classification, and recommender although they are occasionally utilized for computational linguistics. DCNNs' strength comes from their ability to stack. A deep CNN uses a 3D neural network to process the RGB components of a picture simultaneously. In comparison to standard feed-forward neural networks, this significantly reduces the number of neurons necessary to analyze an image.

ConvLSTM is a recurrent neural network having convolutional structures in both the input-to-state and state-to-state phases. It predicts the future condition of a given cell in the matrix using the inputs and previous states of its nearby neighbors.

Mask R-CNN This DNN variation recognizes items in a picture and creates a high-quality segmentation mask for everyone.

3D CNN 3D convolutions use a three-dimensional filter for the data in order to calculate the reduced feature representations (x, y, and z). The outcome is a three-dimensional volumetric area, such as a cube or cube shape. They may be used to identify occurrences in movies, 3D images, and other media. They may well be utilized with two-dimensional inputs like images as well as three-dimensional locations.

3.2 Recurrent Neural Network (RNN)

Reference [31] In CNN, all of the inputs and outputs are distinguishable from each other; nevertheless, in some circumstances, including when trying to forecast the next term of a phrase, the anterior provisions are mandatory, and the preceding words must be remembered because the next word will be ascertained by the previous one.

Working of RNN: RNN generates network connections with loops in them, allowing it to keep track of data. The RNN can take the stream of data thanks to its loop structure.

$$h_t = f\left(W^{(hh)}h_{t-1} + W^{(hx)}x_t\right) \tag{1}$$

$$y_t = softmax\left(W^{(s)}h_t\right) \tag{2}$$

$$J^{(t)}(\theta) = \sum_{i=1}^{|V|}\left(y_{t_i}logy_{t_i}\right) \tag{3}$$

Here, Eq. (1) contains information about the words that came before it in the sequencing. h_t is computed by combining the preceding h_{t-1} vector with the existing x_t word vector. The final accumulation is additionally subjected to a perceptron. In Eq. (2), the softmax function is applied to create a V_1 vector in which all members range from 0 to 1. The parameter of the most expected subsequent word from the lexicon is determined by this probability density function. Equation (3) calculates

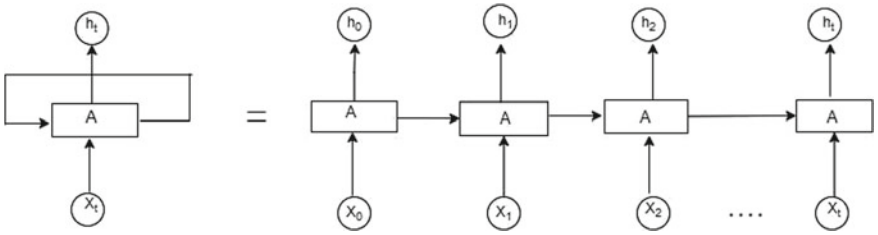


Fig. 3 Recurrent neural network [33]

the loss between the expected and actual word using the gradient descent at each time interval t. The vanishing gradient issue is a fundamental flaw in the sequential model that precludes it from being effective. This indicates that the network has trouble remembering words further down the sequential manner and only generates conclusions based on the most recent occurrences (Fig. 3).

3.3 Long Short-Term Memory (LSTM)

LSTM is a popular sequence model, educated with a set of training sequences. Once qualified, the model is used to perform sequence predictions. Recurrent neural network works superlative when we are dealing with short-term dependencies, but they break down to understand the context behind an input [31]. RNN design has been modified with LSTMs. Long-term and short-term memories may both be stored in LSTM networks. The LSTM's gating structures retain the network's long-term dependencies intact [31]. The gating mechanism allows the network to preserve or release memory and a way to alternatively let information through. They are comprising of a sigmoid neural net layer and a pointwise multiplication operation.

Forget Gate: A forget gate is in charge of eliminating data from the LSTM unit. Through multiplying a filter, information that is not extended for the LSTM to comprehend things or is of fewer relevance is eliminated. This is important for the LSTM efficiency to be optimized.

$$f_t = \sigma\left(W_f\left[h_{t-1,}X_t\right] + b_t\right) \tag{4}$$

Input Gate: The input gate is culpable for using a nonlinear sigmoid function to control what useful elements need to be combined to the cell state.

$$i_t = \sigma\left(W_i\left[h_{t-1}, X_t\right] + b_i\right) \tag{5}$$

Output Gate: The valuation of the next hidden state is ascertained by the output gate. This state reserves data from earlier inputs. It operates with the sigmoid activation

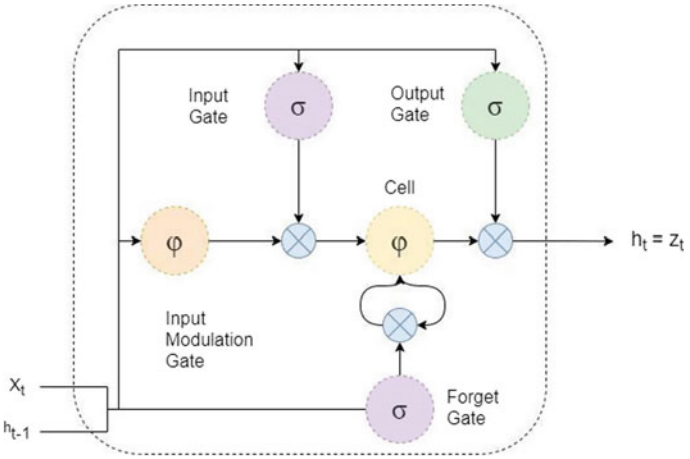


Fig. 4 Long short-term memory network [33]

function, essentially compresses data in the frequency range of 0 to 1, and then multiplies it with cell state data (Fig. 4).

$$O_t = \sigma\left(W_o\left[h_{t-1}, X_t\right] + b_o\right) \tag{6}$$

Image data may also be handled by LSTMs. However, haggling with sequential data such as time series data is their key attribute. For fall detection, a blend of LSTM and CNN is utilized to address general vision-related issues such as picture noise, compression, improper segmentation, orientation, and so on. Fall events may be successfully and efficiently differentiated using LSTMs (Table 1).

4 Challenges and Future Work

4.1 Challenges

The basic challenge of computer vision is object detection which incorporates viewpoint variation, deformation, occlusion, cluttered or textured background, illumination situations, and variability. This literature survey is targeted at numerous CNN versions and few RNN-based variations; however, one can discover different associated learners like LSTM, Bi-LSTM, GRU, and additionally a mixture of CNN with Bi-LSTM or LSTM which may give better accuracy as compared to a single CNN or RNN-based architecture. The combination of multi-dimensional datasets or some set of central persons offers extensive possibilities in research and model overall performance in the real world. The current datasets majorly focus only on indoor

Table 1 Literature review of a paper published from the year 2017–2021

References	Year	Input device	Dataset	Classifier	Performance
[6]	2021	Smartphone, wearable sensors, multimodal sensors	SisFall, UP-fall	Bi-LSTM	Accuracy: 97.41% Accuracy: 97.21%
[7]	2021	Kinect camera	URfall, multiple camera fall	2DCNN + GRU	Accuracy:99.80% Accuracy: 98%
[8]	2021	6 Sensors	Self-collected dataset	AlexNet convolutional neural network	Accuracy: 99.45% Recall: 99.50% F1_measure: 99.48%
[9]	2020	Two HD cameras/frontal and lateral and wearable-based sensors	UP-fall	Mask R-CNN, CNN, LSTM	Accuracy = 100% Precision = 100% Recall 100% F1-score = 100% Loss = 0.02
[10]	2020	IR-UWB monostatic radar	Collected by UWB radar	Combination of a CNN and 1D ConvLSTM layers	Accuracy: 93% Sensitivity: 96.8% Specificity: 92.6%
[11]	2020	MUP6050 and Zigbee-integrated sensor	Sisfall, mobifall	CNN with LSTM (FD-DNN)	Accuracy: 99.17% Sensitivity: 94.09% Specificity: 99.94%
[12]	2020	Monocular camera	Self-collected dataset	Gaussian mixture model (GMM) and convolutional neural network (CNN)	Fall detection rate: 90.5% False alarm rate: 10.0%

(continued)

Table 1 (continued)

References	Year	Input device	Dataset	Classifier	Performance
[13]	2020	Normal camera	FD dataset, Le2i FD dataset, URFD dataset	Yolo, Open pose	**Le2i FD dataset**: Accuracy: 96.91% Precision: 97.65% Sensitivity: 96.51% Specificity: 97.37% F-score: 97.08% **URFD dataset**: Accuracy: 97.33% Precision: 97.78% Sensitivity: 97.78% Specificity: 96.67% F-score: 97.78%
[14]	2020	UWB radar	Self-collected dataset	CNN-LSTM	Accuracy: 98.51%, 96.75%, and 95.48% for positions 1, 2, and 3
[15]	2020	3 types of sensors	Self-collected dataset	Multi-source CNN ensemble (MCNNE) structure	Average Accuracy: 99.30% False-positive rate < 0.69%
[16]	2020	2 Microsoft kinect cameras	URfall dataset	Multi-task hourglass convolutional auto-encoder (HCAE)	Accuracy: 96.2% Sensitivity: 100% Specificity: 93.00% Precision: 92.3% F1_measure: 96.00%
[17]	2020	FLIR ONE thermal camera	Thermal fall dataset	Autoencoder based ConvLSTM network	ROC AUC: 97% PR AUC: 93%
[18]	2020	RGB-D camera	Multiple cameras fall (MCF) dataset, UR fall detection (URFD) dataset and fall detection dataset (FDD)	VGG-16	Accuracy with augmentation: 99.72% Accuracy without augmentation: 96%

(continued)

Table 1 (continued)

References	Year	Input device	Dataset	Classifier	Performance
[19]	2020	Simple camera	URfall & self-build dataset	Mask R-CNN, bi-LSTM	**URfall dataset**: Accuracy: 96.7% F-score: 94.8% Sensitivity: 91.8% Precision: 100% Specificity: 100% **Self-build dataset**: F-score: 94.8% Precision: 98.1% Sensitivity: 92.3%
[20]	2019	Wearable sensors, ambient sensors and vision devices	UP-fall dataset	CNN & LSTM	Accuracy: 96.4% Precision: 84.2% Recall: 81.5% F1_score: 82.3%
[21]	2020	Single camera	FD dataset	CNN with optical flow	Accuracy: 92.84% Precision: 95.27% Recall: 95.42% F1_score: 95.34%
[22]	2019	Multi-bio-radar	Self-build dataset	CNN	Accuracy: 99.29% F-score: 99.28% Sensitivity: 98.57% Precision: 100% Specificity: 100%
[23]	2019	Accelerometer	URFD dataset, Smart watch, and notch dataset	CNN-3B3Conv	**URFD dataset:** Accuracy: 99.86% Precision: 100% Specificity: 100% Sensitivity: 99.72%

(continued)

Table 1 (continued)

References	Year	Input device	Dataset	Classifier	Performance
[24]	2019	Multimodal devices	UCF11 dataset, UCF101 dataset	2D CNN	Accuracy: 92.01% Accuracy: 90.46%
[25]	2018	Camera	Multicam dataset, UP-fall dataset	3D CNN, LSTM	Multicam dataset Accuracy: 95.64% UP-fall dataset Accuracy: 82.26%
[26]	2018	Kinect RGB-D sensor	URFD	Deep CNN, LSTM	Accuracy: 98% Precision: 97% Specificity: 97% Sensitivity: 100%
[27]	2017	Microsoft's Kinect V2 module	NTU RGB + D	Deep CNN	Accuracy: 99.2%
[28]	2017	USB camera	Self-collected dataset	CNN	Accuracy: 94.75%
[29]	2017	Stereo camera	Self-collected dataset	CNN	Accuracy: 91%

environments; the experiments on outdoor environments as well as a few dangerous places are major challenges for deep learning methods that can enhance the lifestyles of age to a few extents. The major challenge in fall detection is of very little elderly data available which ends up in poor accuracy. The solution is to create more elderly data through a few approaches like data augmentation, transfer learning, or creating a new dataset from scratch; then, we can enhance the model accuracy further. The literature survey does not consider any information about biological parameters of a person or health history like Parkinson's disease into the fall detection system. So, one can apply this knowledge to find out the interrelationship between a person's fitness history and his chance of collapse also.

4.2 Future Work

In future, the model performance can be improved after proposing newer learners by modifying existing ones. This will enhance the generalization functionality of the system. We can develop an efficient feature extraction algorithm like YOLOR and YOLOX which gives better precision value and faster inferences compared to YOLOv4 and YOLOv5 [32]. Also, apply ensemble technique by introducing multiple deep learning versions which significantly enhances the overall performance to some extent. With the incorporation of the IoT technology, an implementation of a fall detection system may be developed. Furthermore, the system may be used in a variety of anomalous detection systems. Use the data augmentation strategy to raise the dataset's size to a certain volume, which has a substantial influence on the model's overall performance. Transfer learning can be implemented in case of a lack of a sufficient amount of original datasets. At last, one can also perform experiments on synthetically generated pose data to generalize learners effectively for unrecognized environments and achieve higher precision and recall scores for fall detection.

5 Conclusion

This paper presents an outlook of fall detection systems that use deep learning-based methods. We have reviewed various CNN-based and RNN-based methods. Fall detection systems have almost all been developed using CNN in similar works. The evaluated fall detection system's ambition is to exert the best deep learning methods to predict older people's falls with high accuracy and low false alarms. Various efficient fall detection methods based on a relatively basic and computationally efficient collection of features taken from a publicly available dataset were investigated in this paper. The collected characteristics are utilized to train and evaluate multiple models, with the optimal outcomes coming from a fusion of CNN and RNN, which has an accuracy of about 98 percent. We expect that the information offered in this study will aid researchers in the future development of precise and reliable deep learning-based fall detection systems.

References

1. Rubenstein LZ (2006) Falls in older people: epidemiology, risk factors and strategies for prevention. Age Ageing 35:ii37–ii41
2. World Health Organization (2008) WHO global report on falls prevention in older age; OCLC: Ocn226291980. World Health Organization, Geneva, Switzerland
3. Wild D, Nayak US, Isaacs B (1981) How dangerous are falls in old people at home? Br Med J (Clin Res Ed) 282:266–268
4. Lord S, Smith S, Menant J (2010) Vision and falls in older people: risk factors and intervention strategies. Clin Geriatr Med 26:569–581
5. Zigel Y, Litvak D, Gannot I (2009) A method for automatic fall detection of elderly people using floor vibrations and sound proof of concept on human mimicking doll falls. IEEE Trans Biomed Eng 56(12):2858–2867
6. Waheed M, Afzal H, Mehmood K (2006) NT-FDS—a noise tolerant fall detection system using deep learning on wearable devices. Sensors 2021
7. Sultana A, Deb K, Dhar PK, Koshiba T (2021) Classification of indoor human fall events using deep learning. Entropy 23:328
8. Mobsite S, Alaoui N, Boulmalf M (2020) A framework for elders fall detection using deep learning. 2020 6th IEEE congress on information science and technology (CiSt), pp 69–74
9. Alarifi A, Alwadain A (2021) Killer heuristic optimized convolution neural network-based fall detection with wearable IoT sensor devices. Measurement 167:108258
10. Ma L, Liu M, Na Wang L, Wang YY, Wang H (2020) Room-level fall detection based on ultra-wideband (UWB) monostatic radar and convolutional long short-term memory (LSTM). Sensors 20:1105
11. Liu L, Hou Y, He J, Lungu J, Dong R (2020) An energy-efficient fall detection method based on FD-DNN for elderly people. Sensors 20:4192
12. Yao C, Hu J, Mi W, Deng Z, Zou S, Min W (2020) A novel real-time fall detection method based on head segmentation and convolutional neural network. J Real-Time Image Process
13. Wang B-H, Yu J, Wang K, Bao X-Y, Mao K-M (2020) Fall detection based on dual-channel feature integration, vol 8. IEEE Access
14. Maitre J, Bouchard K, Gaboury S (2020) Fall detection with UWB Radars and CNN-LSTM architecture. J Biomed Health Inf 2168–2194
15. Wang L, Peng M, Zhou Q (2020) Pre-impact fall detection based on multi-source CNN ensemble. IEEE Sens J 20(10)
16. Cai X, Li S, Liu X, Han G (2020) Vision-based fall detection with multi-task hourglass convolutional auto-encoder, vol 8. IEEE Access
17. Elshwemy FA, Elbasiony R, Saidahmed MT (2020) A new approach for thermal vision based fall detection using residual autoencoder. Int J Intell Eng Syst 13(2)
18. Khraief C, Benzarti F, Amiri H (2020) Elderly fall detection based on multi-stream deep convolutional networks. Multimedia Tools Appl
19. Chen Y, Li W, Wang L, Hu J, Ye M (2020) Vision-based fall event detection in complex background using attention guided bi-directional LSTM, vol 8. IEEE Access
20. Martınez-Villasenor L, Ponce H, Perez-Daniel K (2019) Deep learning for multimodal fall detection. IEEE international conference on systems, man and cybernetics (SMC)
21. Brieva J, Ponce H, Moya-Albor E, Martínez-Villaseño L (2019) An intelligent human fall detection system using a vision-based strategy. IEEE explore
22. Anishchenko L, Zhuravlev A, Chizh M (2019) Fall detection using multiple bioradars and convolutional neural networks. Sensors 19:5569

23. Santos GL, Endo PT, Monteiro KHC, Rocha ES, Silva I, Lynn T (2019) Accelerometer-based human fall detection using convolutional neural networks. Sensors
24. Espinosa R, Ponce H, Gutiérrez S, Martínez-Villaseñor L, Brieva J, Moya-Albor E (2019) A vision-based approach for fall detection using multiple cameras and convolutional neural networks: a case study using the UP-Fall detection dataset. Comput Biol Med 115:103520. ISSN 0010-4825
25. Lu N, Wu Y, Feng L, Song J (2018) Deep learning for fall detection: 3D-CNN combined with LSTM on video kinematic data. IEEE J Biomed Health Inf 2168–2194(c)
26. Abobakr A, Hossny M, Abdelkader H, Nahavandi S (2018) RGB-D fall detection via deep residual convolutional LSTM networks. IEEE 978-1-5386-6602-9
27. Tsai TH, Hsu CW (2017) Implementation of fall detection system based on 3D skeleton for deep learning technique. IEEE
28. Solbach MD, Tsotsos JK (2017) Vision-based fallen person detection for the elderly. IEEE international conference on computer vision workshops (ICCVW)
29. Yu M, Gong L, Kollias S (2017) Computer vision-based fall detection by a convolutional neural network. Proceedings of the 19th ACM international conference on multimodal interaction, pp 416–420
30. Mandal M. Introduction to convolutional neural network. Available online: https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/. Last accessed 1 May 2021
31. Dongse N. A guide to RNN: understanding recurrent neural networks and LSTM networks. Available online: https://builtin.com/data-science/recurrent-neural-networks-and-lstm. Last accessed 17 August 2021
32. Colah's blog, understanding LSTM networks, Available online: https://colah.github.io/posts/2015-08-Understanding-LSTMs/. Last accessed 27 August 2015
33. Wang C-Y, Yeh IH, Liao H-Y (2021) You only learn one representation: unified network for multiple tasks

A Deep Learning-Based Approach for Pin-Pointing DNA-Binding in Protein Mutations

Sajan Kumar, Sarvesh Shrof, Sobin C. C, Sunil Kumar, and Geevar C. Zacharias

Abstract Proteins play a very important part in various vital roles such as understanding a disease and are most commonly used in drug formulation. The process of building DNA-binding proteins that is an ideal combination of deliverability, specificity, and activity is not yet fully solved, nevertheless current platforms offer unique advantages, offset by behaviours and properties over each other. In this paper, we propose a method using deep learning which we can predict and pin-point the exact DNA-protein binding site and their conformation due which a particular decease has risen in a person. The proposed method is able to achieve an accuracy of 89.21% compared to some of the existing methods.

Keywords Computational geometry · Graph theory · Hamilton cycles

1 Introduction

Proteins are macro-molecules which are essential for the survival of any living organism. There exist hundreds of thousands of proteins and each one of them vary in their biological functions. Understanding the functionalities of proteins is a crucial biological task, as it can shed light to the cause and cure of numerous diseases. Moreover, many proteins function as disease biomarkers. There are other category of proteins termed as essential proteins, without which no organism can survive [1].

Proteins interact with many other bio-molecules such as DNAs, other proteins, and RNAs. The study of such interactions is a key biological task as it leads to stronger understanding of the entire biological system. The functions of proteins

S. Kumar · S. Shrof · S. C. C (✉)
Department of Computer Science, SRM University AP, Amaravati, Andhra Pradesh, India
e-mail: sobincc@domain.com

S. Kumar
Department of Computer Science, VIT Vellore, Vellore, Tamil Nadu, India

G. C. Zacharias
Department of Computer Applications, MES College of Engineering, Kuttippuram, Kerala, India

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_33

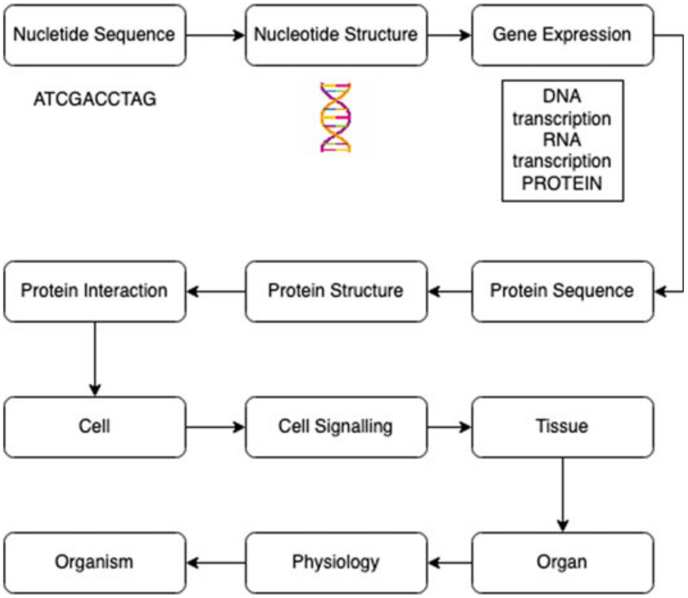


Fig. 1 Flow in biological systems

can be predicted by analyzing their interactions as well. In general, the function annotation of proteins is very important and fundamental mission in cellular-level biology. Protein-DNA interactions play vital roles in a variety of biological process and cellular activities including transcription, gene regulation, DNA repair, DNA replication, DNA recombination, and DNA packaging [2, 3]. The function annotation of proteins is driven by the knowledge of the protein-DNA-binding site. However, the sequence mutations occurring at the binding sites can potentially influence the functions of the resulting proteins as well as may lead to diseases due to the complete abolishment of protein functions (Fig. 1).

Proteins are macro-molecules which play vital roles such as repairing damaged cells, reproducing new cells, etc. Proteins are made up of polymers of structural units called amino acids. There exist many types of amino acids but mostly all of the general purpose proteins are made up of 20 different amino acids. They are the most important molecules in the processes for organism to function and reproduce properly. Their use in medical therapy and drug formulation requires protein is isolation in pure form. Proteins are also commonly used in drug formulation.

The proteins in their pure form can help us understand the target location where the correct medicine needs to act and cure a disease. The receptor proteins are the most common targets amongst receptor proteins and enzymes. There are various types of receptors involved in target estimation. G protein-coupled receptors are the largest group of membrane receptors that is used most frequently in target estimation.

Beta-propeller protein-associated neuro-degeneration [4] (BPAN) is a recognizable NBIA disorder based on the clinical features, the MRI pattern, and the natural history of the disease. Such type of mutation in protein leads to disease whilst binding with various other proteins and molecules inside a person [5].

Recently, various machine learning techniques have evolved for predicting harmful mutations of the human genome [6]. Some of the examples are neural networks [6], support vector machines (SVM) [7], random forest [8], etc. Point-cloud models [9] are used in predicting the effect of the mutations on health of a human.

In this paper, we proposed a method, which uses deep learning method to predict disease-associated missense mutation. Then, we will query databases related to protein bindings such as ClinVar [10], Uniprot and humsavar [11], and the PIR International Protein Sequence [12] Database to extract diseases related to the binding site. After that we extract the sequential and spatial features from the data banks to train our model the deep Alpha [13] + MC-CNN [14].

2 Background

2.1 Nucleotide Sequence and Structure

Nucleotides play a crucial role in the metabolism of the organism at the cellular level. They are fundamental building blocks of all life forms on earth. They are responsible for providing various forms of energy to different parts of the cell many cellular functions, such as amino acid, protein and cell membrane synthesis, moving, cell division, and maintaining cells. There are fundamentally 5 different types of energy that are NTP, ATP, CTP, UTP, and GTP [15].

They are organic molecules made up of nucleoside and a phosphate group. It serves as monomeric units for the construction of nucleic polymers DNA and RNA. These polymers are very essential bio-molecules for an organism survival. These molecules consist of three distinct sub-unit chemical molecules: **phosphate group** and **five-carbon sugar** (deoxyribose or ribose) known as nucleobase. There are 5 types of nucleobases present in all cells known as adenine, guanine uracil, cytosine, and thymine and are represented by A, G, U, C, and T, respectively, whilst writing gnome sequence. DNA consists of A, G, C, and T, whereas in RNA, T is replaced with U. This result in the different structure of RNA and DNA, where A and T form two hydrogen bonds, and G and C form 3 hydrogen bonds resulting in 3-dimensional helix structure. When nucleotides are assembled in a polymeric chain forming a macromolecule are called as nucleic acid. The nucleotides are also known as monomeric units of nucleic acids.

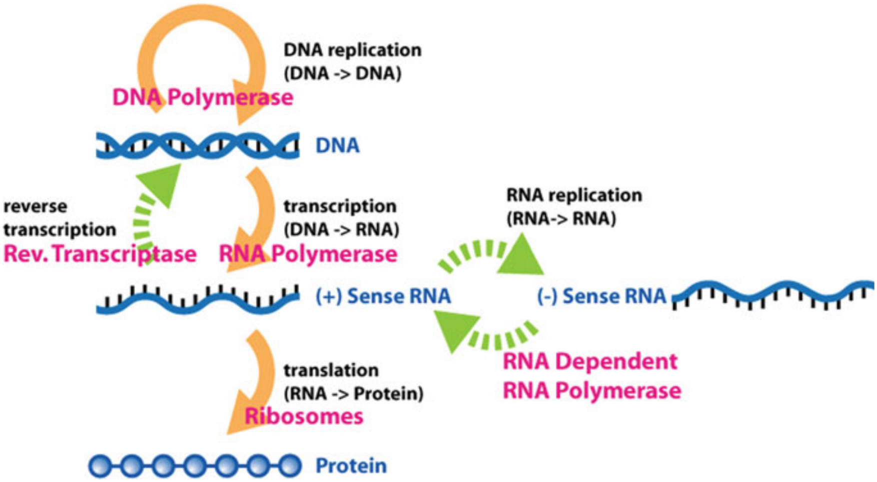


Fig. 2 Extended central dogma with enzymes

2.2 Gene Expression

Gene expression is the process by which the nucleotide molecules are instructed together to form DNA, RNA, and finally protein. It is one of the most important processes in which the first functional functions of an organism are made. The ultimate product is the protein which is the building block of every living organism to reproduce and maintain its life cycle.

Gene expression process production method is used by all known life forms from eukaryotes (multicellular organisms) to prokaryotes(single cell organisms) and even viruses for multiplication and survival. These processes of gene expression give rise to phenotype which is an observable traits in organisms. These traits are the result of protein synthesis which controls the organism's development and structure [16] (Fig. 2).

In DNA, there are two nucleotides oriented in opposite directions. This allows a complementary and base pairs between all the nucleotides molecules. All of these pairings are necessary for transcribing replicating the encoded information. Because of the bonds between the two strands they form a structure name double helix. There are majorly 8 steps are need to be done before it can produce protein as its product. Which is in the following sequence: ***Transcription→ mRNA Processing→ Non-Coding RNA maturation→ RNA export→Translation→Folding→ Translocation→ Protein transport***.

3 Methodology and Implementation

From the very beginning, understanding and exploring the protein structures and their surface interaction with other molecules were an important research topic. Earlier studies showed that due to missense generally neuro-degeneration protein folding, these neuro-degenerative diseases occur. These diseases include Alzheimer's, Parkinson's, Gaucher's disease, and many more. These diseases have seen a higher trend in the death rate in the last 2 decades. Almost all of them have seen an increase of up to 2 fold in the death rate. Protein interaction is essential for all organisms to function properly. With the recent advancement in deep learning, it is now possible to build highly sophisticated models that can ease the visualization and can help us understand what is the pain point where the particular disease has originated from. Various algorithms tried to predict these harmful mutations in human genome due to DNA-binding protein. For example, information theory [17]; Stochastic; Enumerative; Deterministic approach such as Gibbs sampler [18], PhyloGibbs [19], MEME [20], ChiPMunk [21], SeSIMCMC [22], Consensus [23] were some of the methods used to predict the effect of binding sites mutations. Recent advancement in sequencing technology and machine learning approaches such as random forest [24], neural network [25], SVM [26], and Bayes classifies [27] were used to predict affect of protein-DNA interaction. However, in this paper, we propose a new method which integrates both spatial features as well as sequential features of DNA and protein surface binding sites into a Alpha fold [13] + multi-channel convolution neural network (MC-CNN) [14]. We are using Alpha fold [13] to extract the 3-dimensional features of protein and MC-CNN [14] for combing for spatial and sequential features together to predict mutations.

3.1 Collection and Pre-processing of Data

We are integrating databanks such as UniProt [11], RCSB Protein Data Bank [28], and ClinVar [10] to find mutations associated disorders that occur due to DNA-protein binding sites. These databanks have information about sequence of amino acid motifs which is used to generate 3D shapes of proteins, tfs, and complex assemblies of the protein. These comprehensive and high-quality resources will provide information about protein sequence and their functional information in specification of diseases associated mutations. After the collection of the database, we then pre-process data and initialize the data methods on protein structure. These disease-associated mutations will be used as binary class for training samples for training the model. After the prediction, binding sites we perform statistical analyzes for studying disease-associated mutations.

Table 1 Surface interactions sites of disease-associated mutations in protein[4]

Location	Count (frequency)
Buried	2211 (0.22)
PPI	726 (0.12)
Ligand-binding	714 (0.12)
Metal ion-binding	174 (0.029)
DNA/RNA-binding	**420 (0.07)**
Geometric pocket	2177 (0.36)
Other (exposed)	2231 (0.427)
Total	**6025**

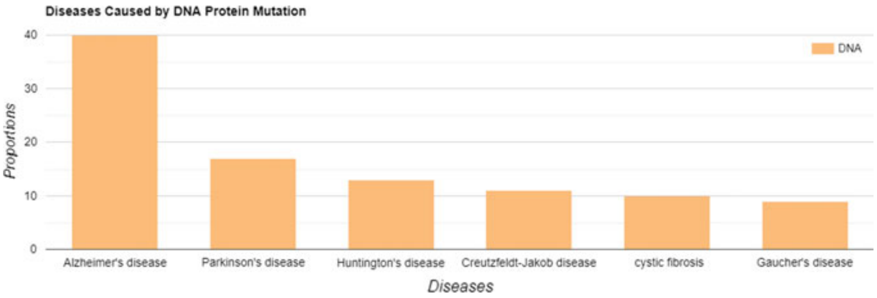


Fig. 3 Proportion of diseases caused by DNA-protein mutation

3.2 *Extraction of Diseases Associated Mutations*

We integrated various databases to collect disease-associated mutations that occur due to DNA-binding with proteins at wrong surface interaction site. Research showed that conformation changes 3-dimensional structures are associated with different types of disease. Specifically, these DNA-binding proteins are the cause of most of the neuro-degenerative diseases [4] such as Alzheimer's, Huntington's, Creutzfeldt-Jakob, Parkinson's, cystic fibrosis, and Gaucher's diseases. Our exploration of dataset we found a total of around 1500 unique diseases associated with mutations. Amongst them around 200 were related cause of DNA-binding proteins [29]. The study in the paper [30] shows the mutations at these sites have a major role in most of the neuro-degenerative disorders. Table 1 shows the distribution of type of binding site and frequency of these disease-associated mutations (Fig. 3).

We investigated many articles where we found out that there are various types of amino acid mutations. These mutated amino acid motifs can be classified into subgroups based on the affinity of size, interaction sites, hydrophobicity, donor or acceptor of hydrogen atom. We used ImMunoGeneTics(IMGT) [31] information system to annotate the amino acids in our sample dataset.

Table 2 Result comparison of different methods with our proposed method

Method	Type	Input	Output	Accuracy
DP-Bind [32]	Sequence based	Protein sequence	0, 1 residues	77.2
DNABINDPROT [33]	Sequence based	Protein sequence	Binding residues	83.3
BindN+ [34]	Sequence based	Protein sequence	Binding residues	79.0
DBindR [35]	Sequence based	Protein sequence	Binding residues	91.41
iDNA-Prot|dis [36]	Sequence based	Protein sequence	Binding residues	72
DR_bind [5]	Structural based	Structure	Binding residues	90.47
DBD-Hunter [37]	Structural based	Structure	Structure	98
Our [Alpha + MCCCN]	**Strutural based**	**Structure**	**Binding residues**	**89.21**

Table 2 compares various different models accuracy as well as input and output. It also suggests that overall the structured based models perform better than sequence-based model for predicting binding motifs.

3.3 *Spatial and Sequential Feature Extraction*

Feature selection plays an important role in any machine learning model for identifying patterns correctly and efficiently. In our proposed method, we have two types of features: spatial and sequential. For feature selection, we used Alpha fold for predicting protein folding and 3DinSight database for selecting tools and visualization of proteins and understanding features required. We used RasMol and virtual reality modelling language (VRML) for displaying structural data derived from the protein data bank (PDB) real-time 3D images. PROSITE and Protein Mutant Database were used for identifying functional domains and mutations. All these features were classified as spatial features and sequential features as input for training our proposed method.

Spatial Features We used thermodynamic parameters and binding data from the ProNIT database to build spatial features. These features were then specified via grid maps using AutoGrid. Initially, we converted PDB files to PDBQT which contained information such as ions, charges, and polar atoms. We used the XGBoost feature selection technique for selecting optimal features for generating spatial features. After selecting the optimal feature set, we use these 6 different probes to generate grids; namely buffers, pH, free energy, binding constant ex, Ka, Kd for wild and mutant entities, etc., carbon atom type, and donation or acceptance of hydrogen. The generated grids are points surrounding the DBP site in the 3-dimensional structure. The grid size is $30 \times 30 \times 30\,\mathrm{A}\hat{3}$ and spacing between each point is 2A. Hereafter, we calculate pairwise interaction energy between each probe and the DBP binding site.

Sequential Features After generating spatial features, we used physicochemical properties of amino acids and links to protein-nucleic acid complex structures, and descriptions about conformational changes of protein and nucleic acid upon binding to train our proposed model. We used these features as input in our proposed method along with metadata. As for the metadata, we defined mutated amino acid and nucleic acid at the binding site as a binary class. We included 10 meta-features along with sequential features identified above for training our model. Namely; original vs mutated amino acid and nucleic acid, type of interaction (direct or indirect), protein structure (alpha, beta sheets), binding site, surface energy, hydrophobicity. Some of the features were categorical so we converted them to numerical values using the one-hot encoding technique. We use these numerical values as features as input in our proposed method.

3.4 Deep Learning Model and Architecture

After extraction of sequential and spatial features for each binary class. For sequential features, we used one-hot encoding for converting it into binary class. We then feed these features as input to the deep learning model in alpha fold to accurately predict 3D structure of protein and use it as input for next model MC-CNN. This type of ensemble model is able to perform better as it increases information retrieval and mapping time low performing overall better than all other methods proposed before.

We finally trained our model using the spatial and sequential features extracted from the datasets. As this is a supervised learning approach, we used spatial features and sequential features for the labelled training dataset. The spatial features were fed into the 3D Conv channel in the MC-CNN model, where we fed grid maps to the channel in our proposed method. Then, we used depthwise convolution and max avg pooling layer. Our proposed method has 3 stages where we use the ensemble method to combine 3 powerful techniques to produce better results. In the first stage, we use alpha fold with AutoGrid to generate protein sequences that are generated from the PDB dataset to the PDBQT format which contains information about the charge and atoms. We then use the generated sequential data along with the metadata to find the binding site. After getting the binding sites, we do statistical analyzes for studying disease-associated mutations associated with the predicted site.

Architecture This model is a supervised learning approach where we used spatial features and sequential features for the labelled training datasets. It has 7 channels out of which 6 spatial features and one sequential feature that we get from the alpha fold. Amongst the 6 spatial features, all different types of thermodynamic parameters-based grid maps are used. The output from the Alpha fold along with autogrid gives us the spatial features that we give to different channels. After normalizing the output, we convert it to a 1-dimensional vector. Output all these 6 channels are concatenated with the 7th channel of sequential metadata that is given as input to a dense neural network. The dense layer in MC-CNN has a rectified Linear activation function to speed up the process of computation. Adam optimizer was used for optimizing the loss function along with the learning rate of 0.0001. To reduce overfitting, weight regularization is used in dense neural networks as well as the convolution layer. After this, we trained the model with a mini-batch size of 10 with 60 epochs to reproduce the result. For further developing a model smaller and computationally fast, we used width multiplier (α) and resolution multiplier (ρ). These hyperparameters help in building smaller and less computational models. Where α and ρ are used for reducing number of channels and input resolution in every layer, respectively. These hyperparameters helped our model in making it computationally light and faster in converging at the result. The trade-off between accuracy and speed can be optimized without losing too much of the performance. For our purpose, we used $\alpha = 0.7$ and $\rho = 0.3$.

4 Performance Evaluation

We used tenfold cross-validation for evaluating our model implemented on 3DinSight, ProNIT, PDB, and Uniprot Humsavar datasets. The receiver operated curve was plotted on tenfold cross-validation to evaluate the performance of out method. We used performance metrics such as Matthew's correlation coefficient (MCC)cite, sensitivity, specificity, accuracy for reporting our results.

4.1 K-Fold Cross Validation

Due to the limited sample dataset available, their was not enough distribution available for all types of DNA-binding protein interaction site available. We introduced method called *k*-fold cross-validation [38] for evaluating our model accuracy over the given limited distribution. We used seven-fold ($k = 7$) cross validation for our model. The results showed us that our models was able to predict disease-associated mutation binding sites in DNA-binding protein surface. We were able predict sites with area under the characteristics curve of 0.91 using both sequential and spacial features. We then performed the same validation method with individual spatial and sequential features alone which resulted in lower AUC of 0.84 and 0.80, respectively. Thus, these individual AUC suggested that spatial features play a crucial role in determining the performance of a model. We then merged the dataset of RNA-based and DNA based data set and evaluated AUC of 0.78 from the Fig. 4. The lower value suggests that the number of sample dataset available is insufficient to train the deep learning model as the environment also affect the binding site surface interaction sites.

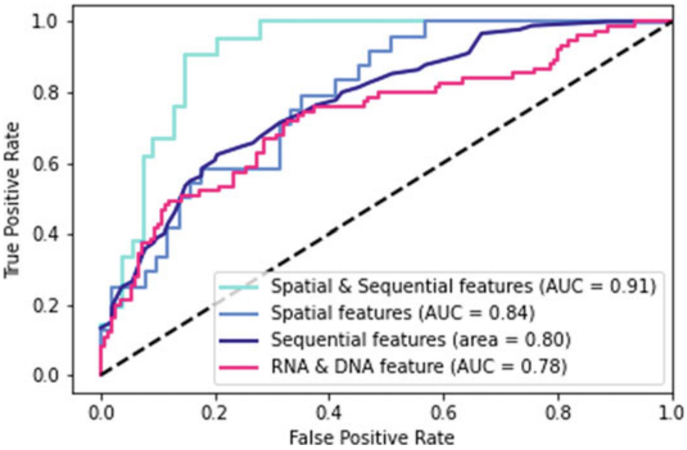


Fig. 4 AUC performance analysis of *k*-fold validation method

4.2 Testing on Unseen Data

We evaluated our final model using splitting dataset into 75:20:5 (% ration) for training, testing, and validation, respectively. We trained our model for best AUC on training dataset followed by evaluation using *k*-fold cross validation and then tested our model on unseen data sample (testing dataset). The overall model accuracy came out to be 0.912134 on the unseen dataset.

4.3 Result Comparison

We used 7 distinct popular methods in Table 2 which we used for comparing our model with. Most of them are sequence-based and few are structural-based methods. As the extraction and report generation time of the implemented method simply the result and accuracy shown here are taken from their original papers. The results shown are based on the average data sizes required to build a comprehensive report from these data sources. The data shown in Table 2 are taken from the original paper and are used for comparison purposes with our model in terms of the type of architecture, input, and output.

4.4 Result Analysis

Based on the above comparative analysis, we can see that structural-based tools and methods are better than sequence-based that uses sequence motifs for finding the binding sites. In structural-based methods, it tries to find the region of interest by comparing binding site three-dimensional structure with the predicted protein structure against the known and experimentally determined interaction sites. The predicted 3-dimensional structure of protein from the alpha fold is matched for surface patches that meet the requirement for Protein-DNA interaction. Nevertheless, sometimes, DNA can change its conformation in the presence of protein which can maximize or minimize protein analogous interface interaction binding. Similarly, protein can also undergo conformations changes. These configuration changes make it difficult to identify binding sites and resulting in false prediction. Whilst predicting the binding site using structures makes it possible for us to compare and find the related mutations and diseases associated with conformation and interface patches.

Our model is based on Alpha fold along with MC-CNN [14] which takes structural input and output and results in the finding of the DNA-binding probe responsible for connecting with protein at specific sites. Our model predicts with an accuracy of 89.21%. Compared to other models, our model is less computational heavy and can compute much faster than compared to other models.

4.5 Conclusions

In our implementation, we found various mutations-related diseases arises from improper binding between DNA and protein. We extracted data from various databanks. We used the extracted sequential and spatial feature to train our deep learning model to predict these binding sites. We were able to correctly predict mutations at surface binding sites and conformations changes. It gives a statistical analysis that produced the result as an affinity toward each binding site with a particular element with amino acid (protein) and the likelihood of mutation that can result in diseases. However, the accuracy of predicting surface binding sites depend on the structure of protein and DNA extracted from database. Our model is not scalable at the moment as this is very specific for three-dimensional structures and sometimes overfitting and also due to restricted access to data. Also, for making this computationally fast, we trade-off with accuracy. Nevertheless, this method can be also be used for development for other applications purposes and development of new drugs.

References

1. Athira K, Gopakumar G (2020) An integrated method for identifying essential proteins from multiplex network model of protein-protein interactions. J Bioinform Comput Biol 18(04):2050020
2. Zhang N, Chen Y, Zhao F, Yang Q, Simonetti FL, Li M (2018) PremPDI estimates and interprets the effects of missense mutations on protein-DNA interactions. PLoS Comput Biol 14(12):e1006615

3. Zhou J, Xu R, He Y, Lu Q, Wang H, Kong B (2016) PDNAsite: identification of DNA-binding site from protein sequence by incorporating spatial and sequence context. Sci Rep 6(1):1–15
4. Gao M, Zhou H, Skolnick J (2015) Insights into disease-associated mutations in the human proteome through protein structural analysis. Structure 23(7):1362–1369. https://doi.org/10.1016/j.str.2015.03.028
5. Gao M, Skolnick J (2008) DBD-Hunter: a knowledge-based method for the prediction of DNA-protein interactions. Nucl Acids Res 36(12):3978–3992. https://doi.org/10.1093/nar/gkn332
6. Koohi-Moghadam M, Wang H, Wang Y, Yang X, Li H, Wang J, Sun H (2019) Predicting disease-associated mutation of metal-binding sites in proteins using a deep learning approach. Nat Mach Intell 1(12):561–567
7. Joachims T (1998) Making large-scale SVM learning practical. Technical report
8. Pal M (2005) Random forest classifier for remote sensing classification. Int J Remote Sens 26(1):217–222
9. Klokov R, Lempitsky V (2017) Escape from cells: deep kd-networks for the recognition of 3d point cloud models. In: Proceedings of the IEEE international conference on computer vision, pp 863–872
10. Landrum MJ, Lee JM, Riley GR, Jang W, Rubinstein WS, Church DM, Maglott DR (2013) ClinVar: public archive of relationships among sequence variation and human phenotype. Nucl Acids Res 42(D1). https://doi.org/10.1093/nar/gkt1113
11. Wu CH (2006) The universal protein resource (UniProt): an expanding universe of protein information. Nucl Acids Res 34(90001). https://doi.org/10.1093/nar/gkj161
12. Berman H, Westbrook J, Feng Z, Gilliland G, Bhat T, Weissig H, Shindyalov I, Bourne P (2020) The protein data bank nucleic acids research. Nucl Acids Res. https://www.rcsb.org/sequence/4Z35
13. Alphafold: a solution to a 50-year-old grand challenge in biology. Deepmind (2020). https://deepmind.com/blog/article/alphafold-a-solution-to-a-50-year-old-grand-challenge-in-biology
14. Koo PK, Ploenzke M (2020) Deep learning for inferring transcription factor binding sites. Curr Opin Syst Biol 19:16–23. https://doi.org/10.1016/j.coisb.2020.04.001
15. Alberts B, Johnson A, Lewis J, Raff M, Roberts K, Walter P, da Veiga AGB (2006) Biologia molecular da Célula. Artmed
16. Alberts B, Johnson A, Lewis J, Raff M, Roberts K, Walter P (2002) Molecular biology of the cell, 4th ed. Garland Science
17. Erill I, Oneill MC (2009) A reexamination of information theory-based methods for dna-binding site identification. BMC Bioinform 10(1). https://doi.org/10.1186/1471-2105-10-57
18. Lawrence C, Altschul S, Boguski M, Liu J, Neuwald A, Wootton J (1993) Detecting subtle sequence signals: a gibbs sampling strategy for multiple alignment. Science 262(5131):208–214. https://doi.org/10.1126/science.8211139
19. Siddharthan R, Siggia ED, Nimwegen EV (2005) PhyloGibbs: a gibbs sampling motif finder that incorporates phylogeny. PLoS Comput Biol 1(7). https://doi.org/10.1371/journal.pcbi.0010067
20. Bailey TL (2002) Discovering novel sequence motifs with meme. Curr Protocols Bioinform. https://doi.org/10.1002/0471250953.bi0204s00
21. Kulakovskiy IV, Boeva VA, Favorov AV, Makeev VJ (2010) Deep and wide digging for binding motifs in chip-seq data. Bioinformatics 26(20):2622–2623. https://doi.org/10.1093/bioinformatics/btq488
22. Favorov AV, Gelfand MS, Gerasimova AV, Ravcheev DA, Mironov AA, Makeev VJ (2005) A gibbs sampler for identification of symmetrically structured, spaced dna motifs with improved estimation of the signal length. Bioinformatics 21(10):2240–2245. https://doi.org/10.1093/bioinformatics/bti336
23. Stormo GD, Hartzell GW (1989) Identifying protein-binding sites from unaligned dna fragments. Proc Nat Acad Sci 86(4):1183–1187. https://doi.org/10.1073/pnas.86.4.1183
24. Ardakani FB, Schmidt F, Schulz MH (2019) Predicting transcription factor binding using ensemble random forest models. F1000Research 7:1603. https://doi.org/10.12688/f1000research.16200.2

25. Heumann GD, Lapedes JM, Stormo AS (1994) Neural networks for determining protein specificity and multiple alignment of binding sites. In: Proceedings. International conference on intelligent systems for molecular biology. https://pubmed.ncbi.nlm.nih.gov/7584389/
26. Pandurangan AP, Blundell TL (2019) Prediction of impacts of mutations on protein structure and interactions: SDM, a statistical approach, and mcsm, using machine learning. Protein Sci 29(1):247–257. https://doi.org/10.1002/pro.3774
27. Hu S, Ma R, Wang H (2019) An improved deep learning method for predicting dna-binding proteins based on contextual features in amino acid sequences. Plos One 14(11). https://doi.org/10.1371/journal.pone.0225317
28. Fermi G, Perutz MF, Shaanan B, Fourme R (1984) The crystal structure of human deoxyhaemoglobin at 1.74 å resolution. J Molecular Biol 175(2):159–174
29. Chaudhuri TK, Paul S (2006) Protein-misfolding diseases and chaperone-based therapeutic approaches. FEBS J 273(7):1331–1349. https://doi.org/10.1111/j.1742-4658.2006.05181.x
30. Le DH (2020) Machine learning-based approaches for disease gene prediction. Briefings Functional Genom 19(5–6):350–363. https://doi.org/10.1093/bfgp/elaa013
31. Ehrenmann F, Lefranc MP (2011) IMGT/DomainGapAlign: IMGT standardized analysis of amino acid sequences of variable, constant, and groove domains (IG, TR, MH, IgSF, MhSF). Cold Spring Harbor Protocols 2011(6). https://doi.org/10.1101/pdb.prot5636
32. Lin WZ, Fang JA, Xiao X, Chou KC (2011) iDNA-Prot: Identification of DNA binding proteins using random forest with grey model. PLoS One 6(9). https://doi.org/10.1371/journal.pone.0024756
33. Wang L, Huang C, Yang MQ, Yang JY (2010) Bindn for accurate prediction of dna and rna-binding residues from protein sequence features. BMC Syst Biol 4(S1). https://doi.org/10.1186/1752-0509-4-s1-s3
34. Chen YC, Wright JD, Lim C (2012) DR_Bind: a web server for predicting DNA-binding residues from the protein structure based on electrostatics, evolution and geometry. Nucl Acids Res 40(W1). https://doi.org/10.1093/nar/gks481
35. Hwang S, Gou Z, Kuznetsov IB (2007) DP-Bind: a web server for sequence-based prediction of DNA-binding residues in DNA-binding proteins. Bioinformatics 23(5):634–636. https://doi.org/10.1093/bioinformatics/btl672
36. Ozbek P, Soner S, Erman B, Haliloglu T (2010) DNABINDPROT: fluctuation-based predictor of DNA-binding residues within a network of interacting residues. Nucl Acids Res 38(suppl_2). https://doi.org/10.1093/nar/gkq396
37. Ding XM, Pan XY, Xu C, Shen HB (2010) Computational prediction of DNA-protein interactions: a review. Curr Comput Aided-Drug Des 6(3):197–206. https://doi.org/10.2174/157340910791760091
38. Cristianini N (2004) Cross-validation (k-fold cross-validation, leave-one-out, jackknife, bootstrap). Dictionary Bioinform Comput Biol. https://doi.org/10.1002/9780471650126.dob0148.pub2

Outbreak Prediction of COVID-19 3rd Wave in Asia

Avinash Sharma, Dharminder Yadav, L. S. Reen, and Umesh Chandra

Abstract Coronavirus is a pandemic for whole world and infected more than 200 countries of the world. Spreading of coronavirus started from china at the end of December and within three months, it infected whole world. Coronavirus is belonging to beta coronavirus family. Common symptoms of coronavirus are fever, dry cough, fatigue, and respiratory-related problem. This paper tries to study the infection rate of coronavirus in Asian countries, rest of world countries, and overall world countries. Asia is largest continent of the world which contains approximate 50 countries and highest contributes in GDP. Population of Asian countries 446.27 crore, that is 60 percentage of whole world population and covers 30 percent geographical area of world. China and India are the most populated countries in Asia. Total confirmed cases 2,056,051, recovered cases 502,045, and death cases 134,177 in the world. Scholar also divides the Asia continent into six regions such East, South, Central, North, Southeast and Western Asia for better understanding infection of coronavirus. This paper analyzes the confirmed cases, recovered cases, and death cases in Asia and understands the infection pattern date wise and countries wise. Machine learning algorithm is used for prediction infection in the world and also predicts future infection rate and death rate in Asian countries and world. Prophet is used for future prediction of confirmed and death cases in the Asian countries.

Keywords Asia · SARS-CoV · COVID-19 · Machine learning · SVM · Prophet · LSTM · Navies Bayes theorem

A. Sharma (✉)
Department of CSE, Maharishi Markandeshwar Engineering College, Maharishi Markandeshwar (Deemed to be University) Mullana, Ambala 133207, Haryana, India
e-mail: asharma@mmumullana.org

D. Yadav
Department of Computer Science and Technology, Glocal University, Saharanpur, UP, India

L. S. Reen
Seth Jai Parkash, Institute of Information Technology, Yamuna Nagar, Radaur, Haryana, India

U. Chandra
Department Computer Science and Technology, Banda University of Agriculture and Technology, Banda, UP, India

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_34

1 Introduction

World is divided into seven continents in which Asia is the largest continent of world [1]. Asia covers total 30% of land area of world which is 44,579,000 km^2 and 8.7% of world total surface area [2]. A lot of oldest civilization developed in Asia such ancient India, ancient China, and Mesopotamia [3]. Asia contribution in the world is highest in the form of gross domestic product (GDP) and purchasing power parity (PPP) [4]. China, Japan, and India are the top three economy of the Asia as of 2018 survey [5]. Asia is bounded by three oceans, from the east by Pacific Ocean, from south by Indian Ocean, and from the north by Arctic Ocean, but there is no clear boundary between Asia and Europe. This paper studies the coronavirus pandemic of the year 2020 in Asia. For better understanding this coronavirus pandemic, scholar divides the Asia continent into six regions such as East, South, Central, North, Southeast, and Western Asia [6wiki]. Coronavirus started from seafood market of Wuhan city of Hubei Provenience China end of December 2019 [1]. This virus found in bats and transmitted to human as well as mammals [2]. World Health organization renames to COVID-19 in February 2020, other name of coronavirus is Human coronavirus, 2019-nCoV, and severe acute respiratory syndrome (SARS-CoV) [3]. Four categories of coronavirus family are alpha coronavirus, beta coronavirus, delta coronavirus, and gamma coronavirus [6]. Coronavirus are six types 229E, OC43, HKU-1, NL-63, severe acute respiratory syndrome coronavirus (SARS-CoV), and middle east respiratory syndrome coronavirus (MERS-CoV), which are responsible for mild and lower respiratory infection [7]. It is ribonucleic acid (RNA) virus, and common symptom of this virus is dry cough, fever, respiratory problem, fatigue, and sputum production, shortness of breath, headache, haemoptysis, diarrhea, and dyspnoea [paper] [5]. This virus spread worldwide and pandemic for the world due to non-availability of medicine and treatment only precaution is the safety measures. Scholar studies the infection rate in different region of Asia, world, and rest of Asia. East Asia contains 8 countries, South Asia contains 8 countries, Central Asia contains 5 countries, North Asia contains one country, Southeast Asia contains 11 countries, and Western Asia contains 20 countries. Paper organized in to five sections, introduction of coronavirus and Asian countries regions; preprocessing and framework of data; data analysis and results and discussion; outbreak prediction in Asia region and conclusion of this paper.

2 Data Preprocessing and Framework

Data contains 20,097 rows and 7 columns such as date, state, country, last update, confirmed, deaths, and recovered and collected from kaggle site. Most of the rows of state columns are NaN and few countries such as USA, Canada, and China state-wise data is available so replaced with white space. Divide the data into six regions East, South, Central, North, Southeast, and Western Asia. East Asia contains

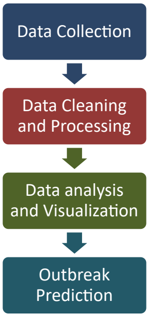


Fig. 1 Methodology for this study

8 countries are China, Mongolia, North Korea, South Korea, Japan, Hong Kong, Taiwan, and Macau. China is the largest country in East Asia in land wise as well as population wise and infection of coronavirus started from China. South Asia contains 8 countries are Sri Lanka, Bangladesh, India, Afghanistan, Pakistan, Bhutan, Nepal, and Maldives. Population and geographical area of India are largest in South Asia region. Central Asia contains 5 countries are Tajikistan, Uzbekistan, Kazakhstan, Turkmenistan, and Kyrgyzstan. North Asia contains one country is Russia and sometime it is called Siberia. Southeast Asia contains 11 countries are Brunei, Cambodia, Indonesia, Laos, Malaysia, Myanmar, Philippines, Singapore, Thailand, Timor Lester, Vietnam, Christmas Island, and Cocoas Islands. Indonesia is the largest economy in this region with GDP of US$932.4 billion [8]. Western Asia contains 20 countries are Georgia, Armenia, Azerbaijan, Turkey, Cyprus, Syria, Lebanon, Israel, Palestine, Jordan, Iraq, Iran, Kuwait, Bahrain, Qatar, Saudi Arabia, and United Arab Emirates. Turkey is the largest economy in the western Asia region followed by Saudi Arabia and Iran [7]. Active case is calculated by death case and recovered cases subtracted from confirmed cases so one new column is added to dataset. Dataset contains the data from January 22, 2020 to April 15, 2020, total of 84 days data. In Asia, total confirmed cases 356,262, recovered 161,744 and death cases 12,205. Rest of the world dataset contains total confirmed, recovered death, and active cases are 1,699,789, 340,301, 121,972, and 1,237,516, respectively, including with 141 countries. Figure 1 represents the methodology of this case study.

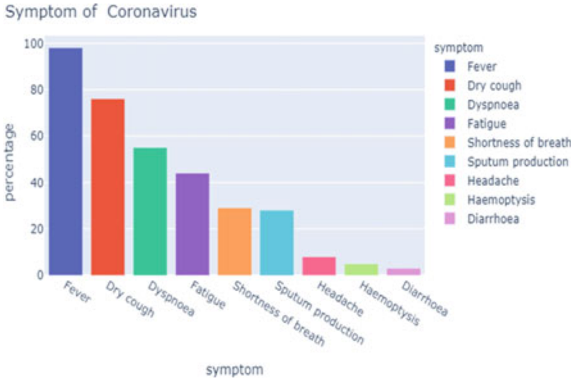
3 Data Anaylsis and Result Discussion

Discuss the analysis of coronavirus effect on Asian regions East, South, Central, North, Southeast, Western Asia and rest of the world. For analysis of data used Python language with package NumPy, Pandas, Keras, fbprophet, sklearn, TensorFlow, Matplotlib, and Plotly. Figures 2 and 3 show the bar and pie chart of coronavirus symptoms, fever and dry cough are the common symptoms [paper]. Figure 4 shows the worldwide confirmed, recovered death, and active case. Till March 18, 2020, confirmed, recovered death, and active cases grow linear, but after 18 March, these cases grow exponentially and reached to 2,056,051 followed by 502,045 recovered cases, 134,177 deaths cases, and 1,419,829 active cases till April 15, 2020. Figure 5 depicts the Central Asia with 3046 confirmed, 425 recovered, 25 death, and 2596 active cases. Figure 6 represents the South Asia confirmed (21,001), recovered (3052), death (598), and active (17,351) cases. Figure 7 depicts Southeast Asia confirmed (22,547), recovered (5971), death (955), and active (15,621) cases. Figure 8 illustrates Western Asia confirmed (182,188), recovered (63,394), death (6708), and active (112,086) cases. Figure 9 represents East Asia confirmed (102,077), recovered (98,685), death (3717), and active (11,575) cases. Figure 10 depicts North Asia confirmed (24,490), recovered (1986), death (198), and active (22,306) cases. Figure 11 represents Asia confirmed (356,262), recovered (161,744), death (12,205), and active (182,313) cases. Figure 12 depicts rest of Asia confirmed (1,699,789), recovered (340,301), death (121,972), and active (1,237,516) cases.

Figure 13 depicts the pairwise correlation between confirmed, recovered death, and active cases in Asian countries. Figure 14 depicts the thematic map of the world with confirmed cases in the Asian countries. Choropleth maps function is used to show the better statistic over a geographical area. The red dark color on the map represents the infection in these countries more than 5000, lighter color, or blue color depicts, less contaminated countries. On clicking on the map, it shows the country name, confirmed case, and deaths cases. East and West Asia are the most contaminated region but Central Asia is less contaminated region. Figure 15 depicts the confirmed cases in the world. Figure 16 depicts the Central Asia country-wise cases, and out of 5 countries, only three countries (Uzbekistan, Kazakhstan, and Kyrgyzstan) data are available and minimum confirmed, recovered, and death cases in this region of Asia. Confirmed cases in Kazakhstan are maximum and in Kyrgyzstan are minimum.

Figure 17 depicts the confirmed, recovered, and death cases in East Asia. Total 8 countries in East Asia but 4 countries (China, Japan, South Korea and Magnolia) data is available. China is the most infected country followed by Japan South Korea,

Fig. 2 Bar chart of coronavirus symptoms



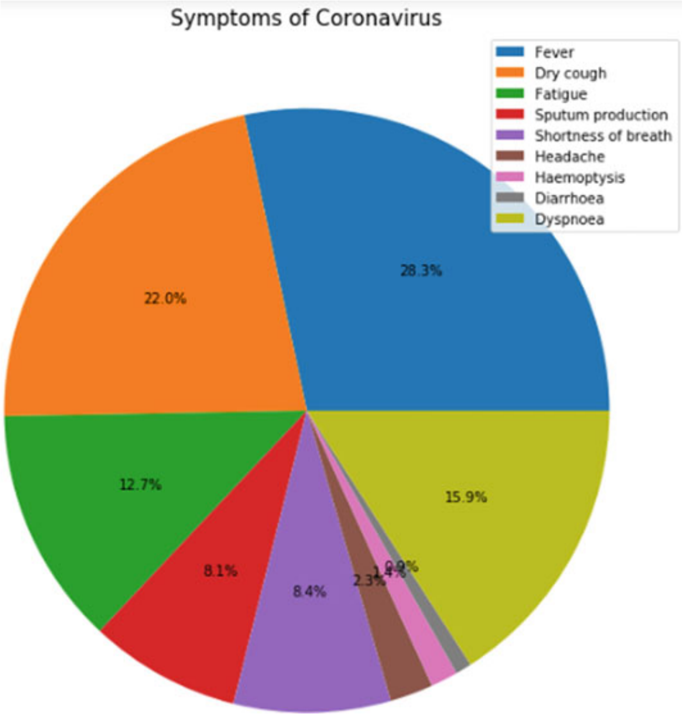
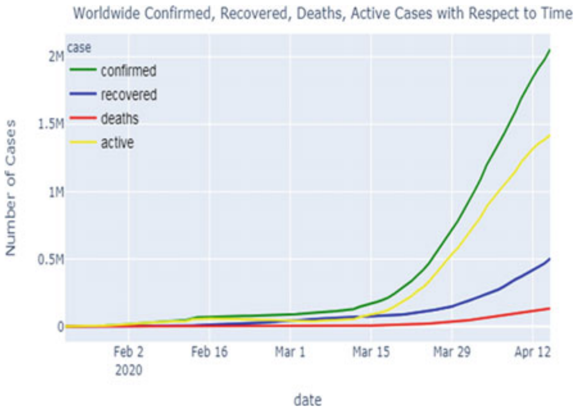


Fig. 3 Pie chart of coronavirus symptoms

Fig. 4 World confirmed, recovered, death, active case



and in Mongolia, confirmed cases are least. Figure 18 depicts the cases in South Asia, India, and Pakistan is the most infected countries; Nepal (16) and Bhutan (5) are the least infected countries. Figure 19 depicts cases in North Asia (Only Russia in North Asia) region. Figure 20 depicts the cases in Southeast Asia, Philippine,

Fig. 5 Central Asia confirmed, recovered, death, active case

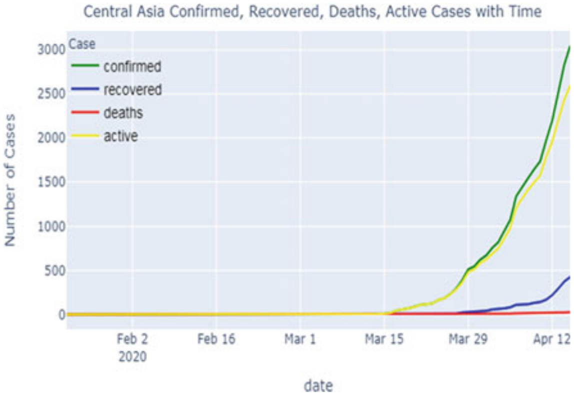


Fig. 6 South Asia confirmed, recovered, death, active case

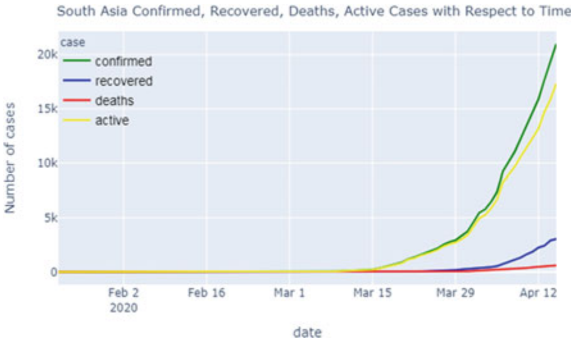
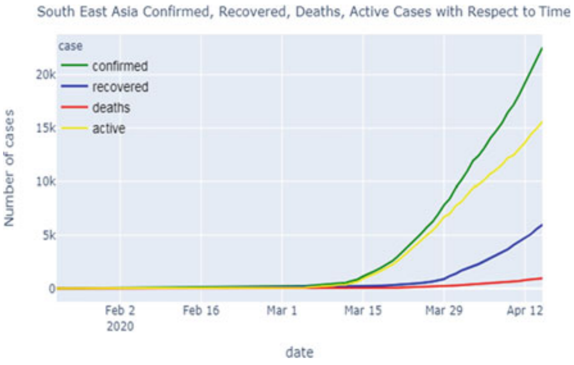


Fig. 7 Southeast Asia confirmed, recovered, death, active case



Indonesia, Malaysia have maximum confirmed case and Brunei (136) and Laos (19) have minimum confirmed cases. Figures 22, 23, and 24 shows the confirmed recovered and death cases in west Asia region. Iran, Turkey, and Israel have highest confirmed cases and Georgia (306) and Syria (33) have least confirmed cases. In Asia region, most infected country is China with 83,356 confirmed cases followed

Fig. 8 West Asia confirmed, recovered, death, active case

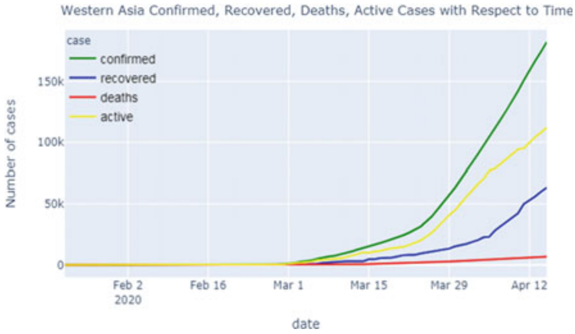


Fig. 9 East Asia confirmed, recovered, death, active case

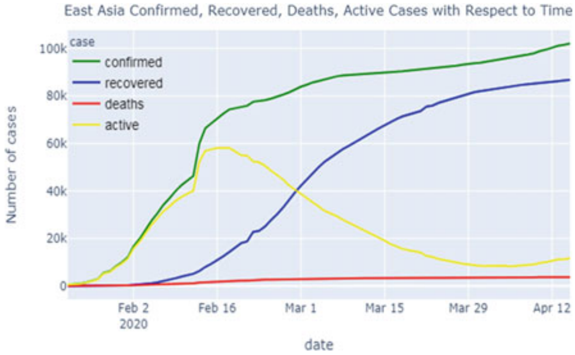
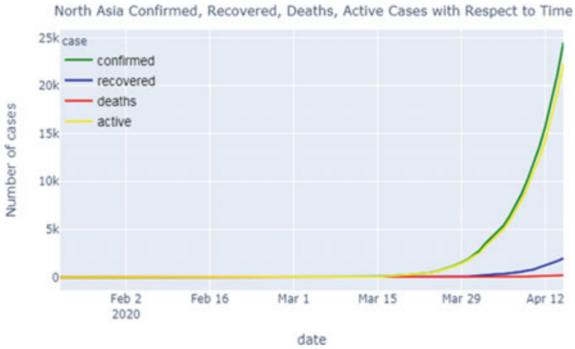


Fig. 10 North Asia confirmed, recovered, death, active case



by Iran (76,389), Turkey (69,392), and India (12,322). Maldives (22), Nepal (16), and Bhutan (5) are the least infected countries in Asia continent. Figures 24 and 25 represent the confirmed and death cases in Asia. West (confirmed 182,188 and death 6708) and East (confirmed 102,077 and death 3717) Asia are the most infected region, and Central (confirmed 3046 and death 25) Asia is the least infected region in the Asia continent. Table 1 gives the recovery rate and mortality rate per 1000 confirmed cases in Asia continent. Mortality rate per 1000 confirmed case of Indonesia (9.13)

Fig. 11 Asia confirmed, recovered, death, active case

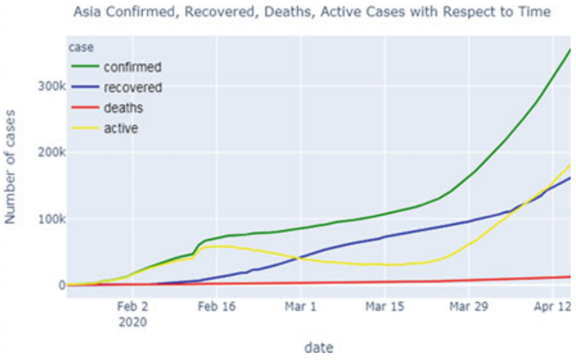


Fig. 12 Rest of Asia confirmed, recovered, death, active case

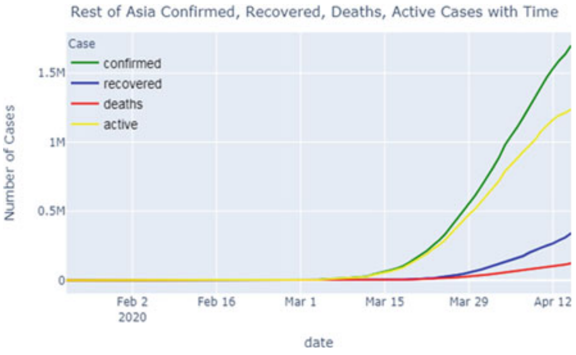
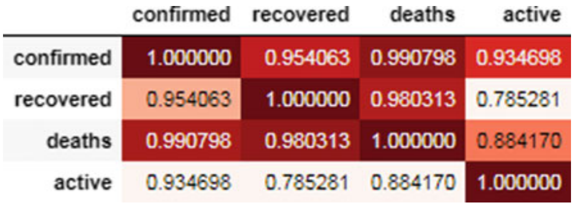


Fig. 13 Correlation b/w confirmed, recovered, death, active case



	confirmed	recovered	deaths	active
confirmed	1.000000	0.954063	0.990798	0.934698
recovered	0.954063	1.000000	0.980313	0.785281
deaths	0.990798	0.980313	1.000000	0.884170
active	0.934698	0.785281	0.884170	1.000000

is highest and Qatar (0.19) is least. Recovery rate per 1000 confirmed cases of China (93.95) is highest and Bangladesh (3.98) is least (Fig. 21).

$$\text{RecoveryRate} = \frac{\text{Death Case}}{\text{Confirmed cases}} * 100 \quad \text{Mortality Rate} = \frac{\text{Death Case}}{\text{Confirmed cases}} * 100$$

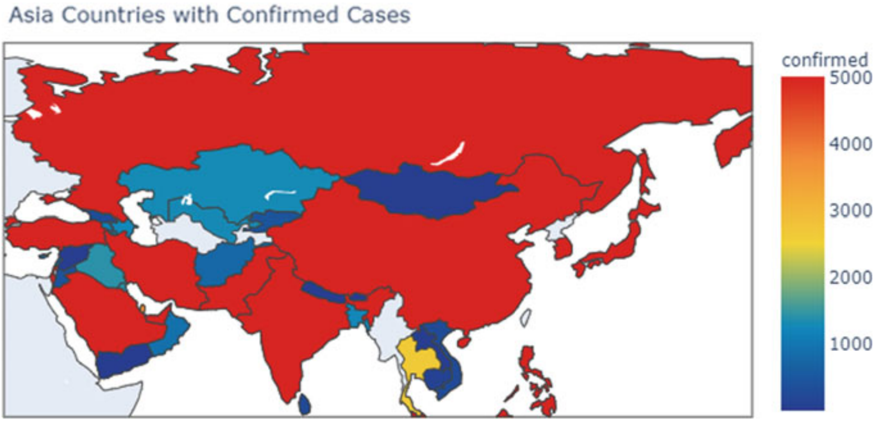


Fig. 14 Country-wise confirmed cases in the Asia

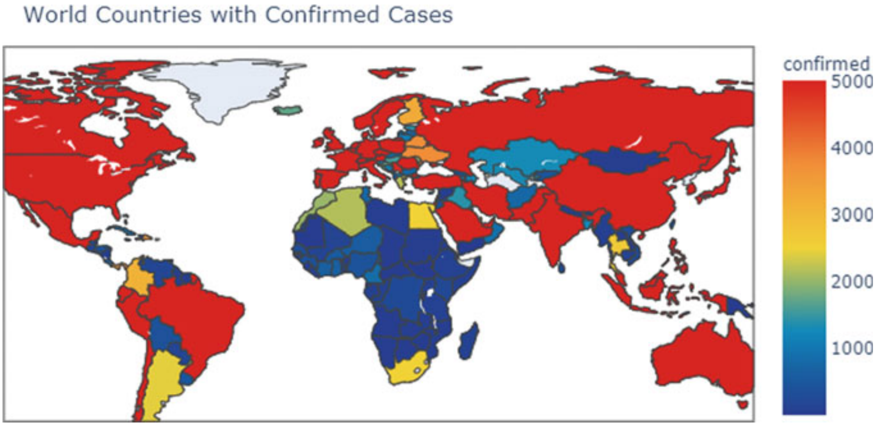


Fig. 15 Country-wise confirmed cases in the world

4 Outbreak prediction for the West Asia, East Asia, and Asia

Prophet is the machine learning algorithm and subpart of artificial intelligence. Prophet is the time series forecasting algorithm for future prediction and implemented in Python and R. It is open-source software developed by Facebook. It is an adaptive model which used nonlinear data to predict for yearly, monthly, and daily excluding holiday. Prophet easily handles missing data and outlier of trend. Prophet is accurate and fast because it used a state-of-the-art platform for statistical modeling that provides forecast in very quickly. In this paper, we used this algorithm for prediction of confirmed and death case for the West Asia region, East Asia region, and Asia

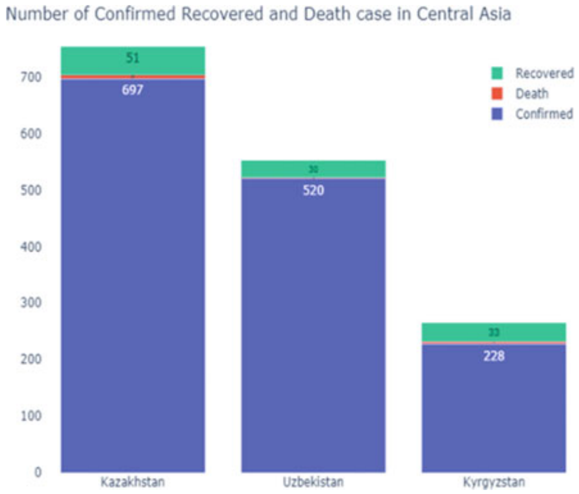


Fig. 16 Central Asia country-wise cases

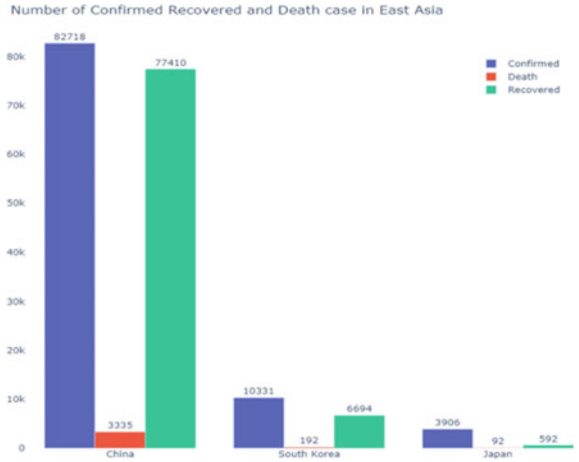


Fig. 17 East Asia country-wise cases

continent for next 10 days given in Table 2. “yhat” represents the predicted value, “lower” and “upper” represent the minimum and maximum value of prediction, “y” represents the actual value. Graphs related to outbreak prediction are given in the appendix (Table 3).

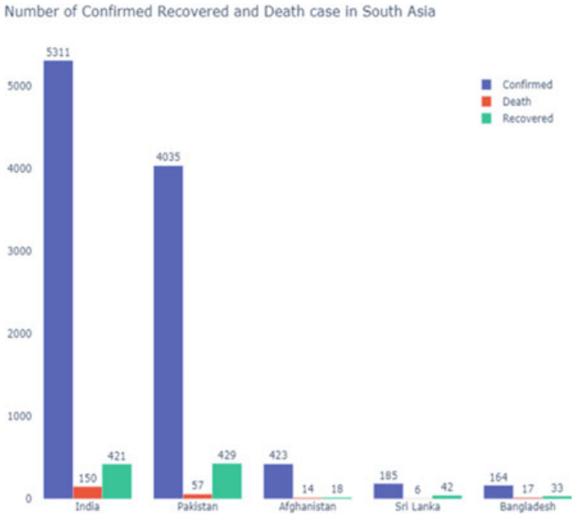


Fig. 18 South Asia country-wise case

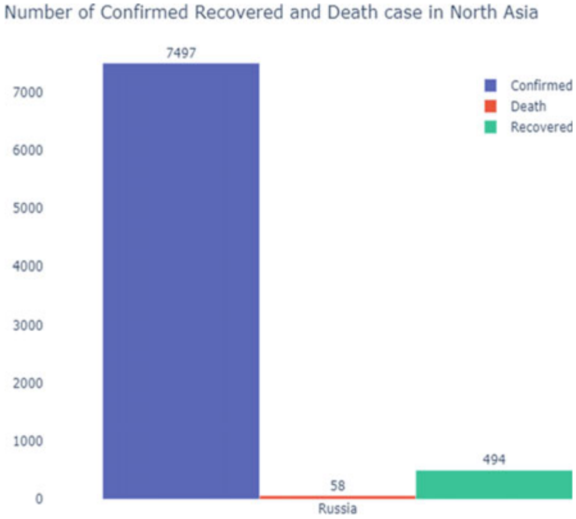


Fig. 19 North Asia country-wise case

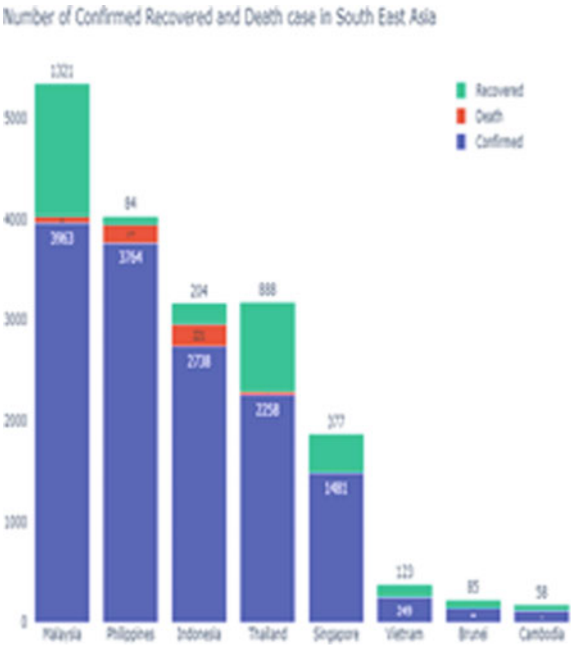


Fig. 20 Southeast Asia country-wise case

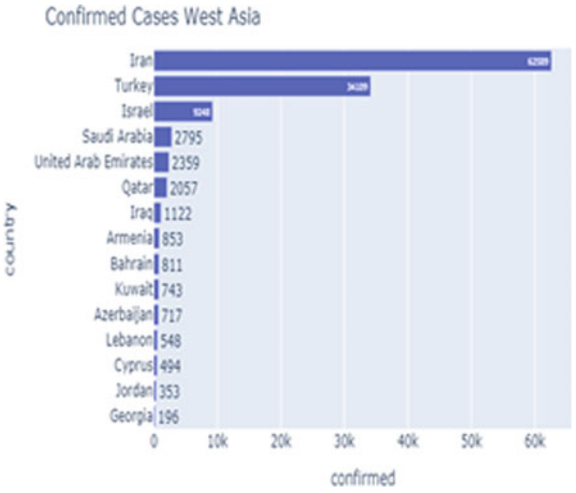


Fig. 21 Confirmed cases in West Asia

Table 1 Recovery and mortality rate per 1000 confirmed cases

S. No.	Country	Confirmed	Deaths	Recovered	Recovery rate %	Mortality rate %
1	Indonesia	5136	469	446	8.68	9.13
2	Philippines	5453	349	353	6.47	6.4
3	Iran	76,389	4777	49,933	65.37	6.25
4	Iraq	1415	79	812	57.39	5.58
5	Bangladesh	1231	50	49	3.98	4.06
6	China	83,356	3346	78,311	93.95	4.01
7	India	12,322	405	1432	11.62	3.29
8	Turkey	69,392	1518	5674	8.18	2.19
9	South Korea	10,591	225	7616	71.91	2.12
10	Japan	8100	146	853	10.53	1.8

Fig. 22 Deaths case in West Asia

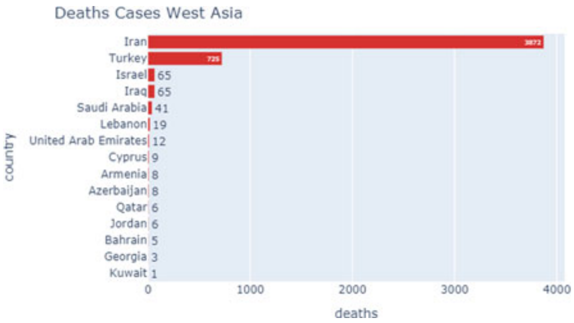
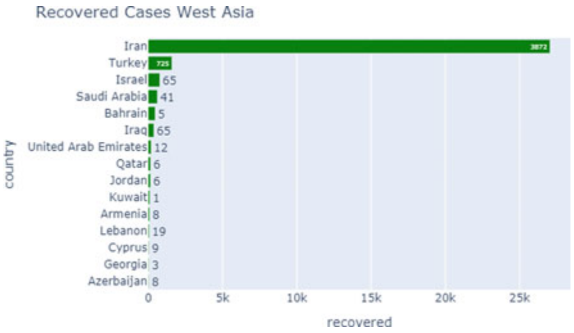


Fig. 23 Recovered cases in West Asia



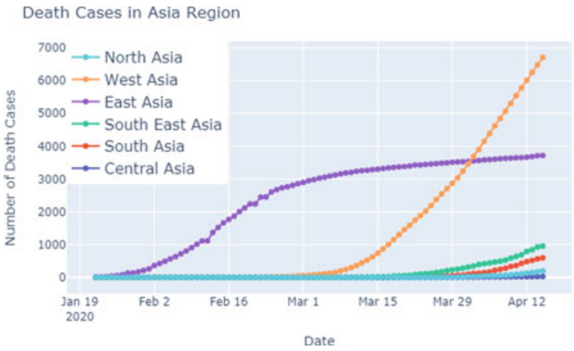


Fig. 24 Confirmed case in Asia

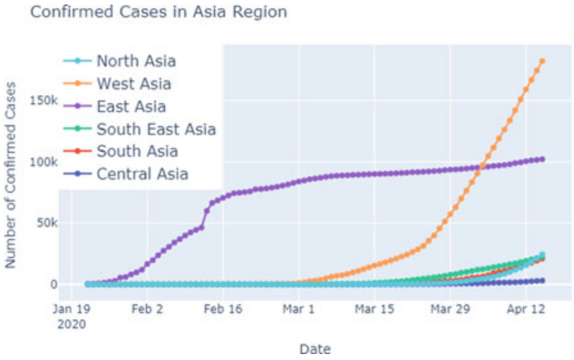


Fig. 25 Death cases in Asia

Table 2 Accuracy with different machine learning algorithm

S. No.	Algorithm	Accuracy in %
1	Decision tree	99
2	Random forest	99
3	Naive Bayes theorem	78

Table 3 Prediction of confirmed and death cases for the US, China, world by using Prophet algorithm

S. No.	Date	Prediction of confirmed cases by using prophet algorithm			Prediction of death cases by using prophet algorithm		
		West Asia	East Asia	Asia	West Asia	East Asia	Asia
1	16-04-2020	187,013.7	101,531.5	355,638.9	6912.164	3727.429	12,323.45
2	17-04-2020	194,594.5	102,052.8	366,930.7	7143.8	3731.966	12,641.39
3	18-04-2020	202,206.2	102,320	378,066.6	7372.194	3753.752	12,977.02
4	19-04-2020	209,730.7	102,692.5	389,479.4	7604.68	3760.924	13,307.55
5	20-04-2020	216,924.1	102,729.1	400,173.7	7827.08	3777.729	13,631.2
6	21-04-2020	224,316.6	102,944.5	411,263.7	8050.682	3790.151	13,954.91
7	22-04-2020	231,587.1	103,447.2	422,524.2	8279.199	3792.569	14,268.31
8	23-04-2020	238,687.1	104,430.2	433,231.9	8509.691	3812.988	14,589.27
9	24-04-2020	246,267.8	104,951.5	444,523.7	8741.328	3817.525	14,907.22
10	25-04-2020	253,879.5	105,218.7	455,659.5	8969.721	3839.312	15,242.84

5 Conclusions and Future Scope

This paper proposed study focuses on analyzing the effect of COVID-19 on the Asia continent. Scholar also analyzes the region-wise infection in Asia and divides Asia continent into five region East Asia, South Asia, West Asia, North Asia, and Southeast Asia. West Asia is having the highest confirmed case, death case, and recovered case. The infection of COVID-19 in Asia continent is lesser than European countries and rest of the world but population and area-wise Asia is largest as compare to rest of world. After China, Iran, Turkey, Israel, India, Pakistan, and Saudi Arabia are the most infected countries in Asia continent. China has the highest recovery rate of 93.95%, and Bangladesh has the least nearly 3.98% till April 2020. Indonesia has the highest mortality rate of 9.13% and Qatar at the bottom 0.19%. Predict the outbreak of COVID-19 in the West Asia, East Asia, and Asia continent by using Prophet time series prediction algorithm. Further, we will try to analyze and future prediction of the outbreak of COVID-19 in the Australia and European continent.

Appendix 1

See Fig. 26.

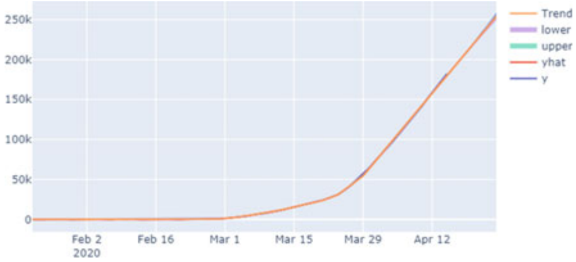


Fig. 26 Predicted confirmed cases in West Asia

Appendix 2

See Fig. 27.

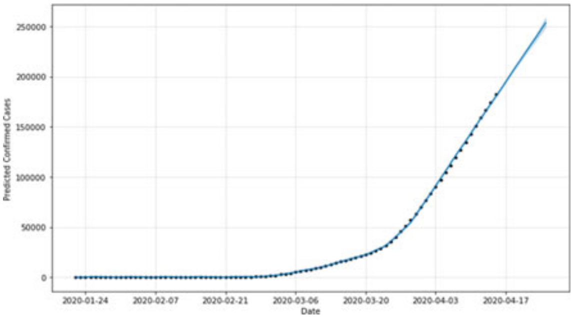


Fig. 27 Predicted confirmed cases in West Asia y and yhat

Appendix 3

See Fig. 28.

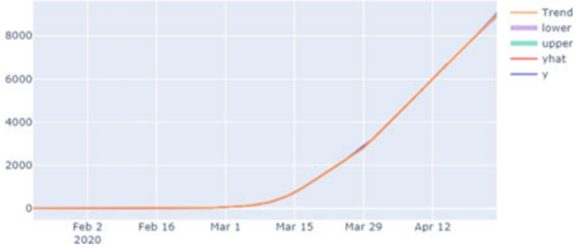


Fig. 28 Predicted death cases in West Asia

Appendix 4

See Fig. 29.

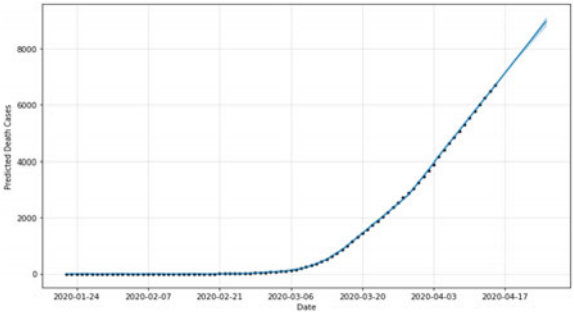


Fig. 29 Predicted death cases in West Asia y and yhat

Appendix 5

See Fig. 30.

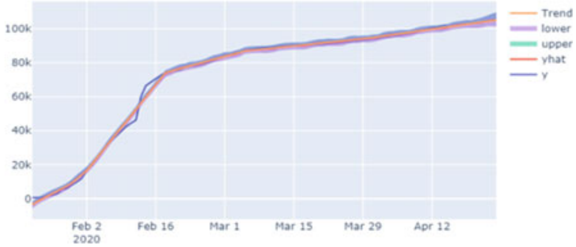


Fig. 30 Predicted confirmed cases in East Asia

Appendix 6

See Fig. 31.

Appendix 7

See Fig. 32.

Fig. 31 Predicted confirmed cases in East Asia y and yhat

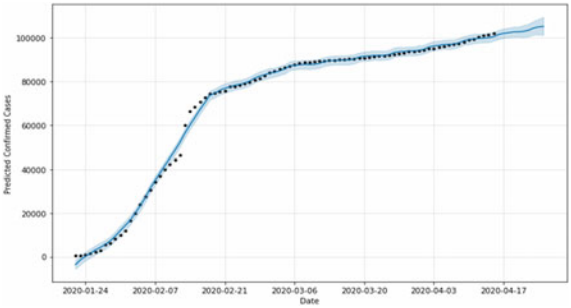
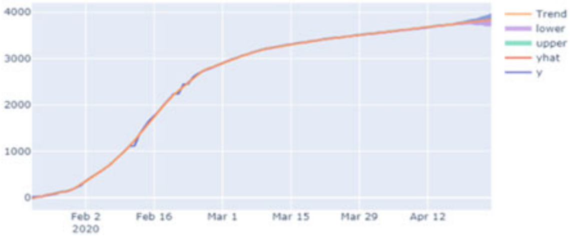


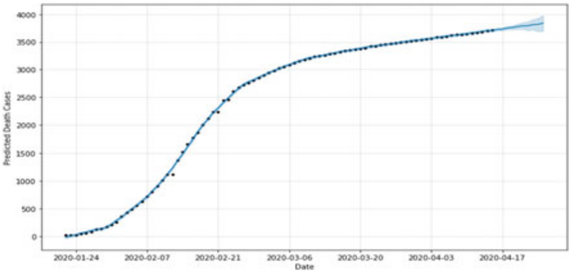
Fig. 32 Predicted death cases in East Asia



Appendix 8

See Fig. 33.

Fig. 33 Predicted death cases in East Asia y and yhat



Appendix 9

See Fig. 34.

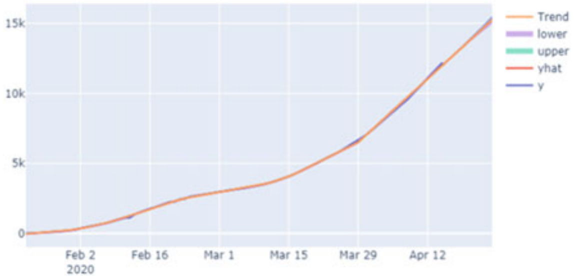


Fig. 34 Predicted death cases in East Asia

Appendix 10

See Fig. 35.

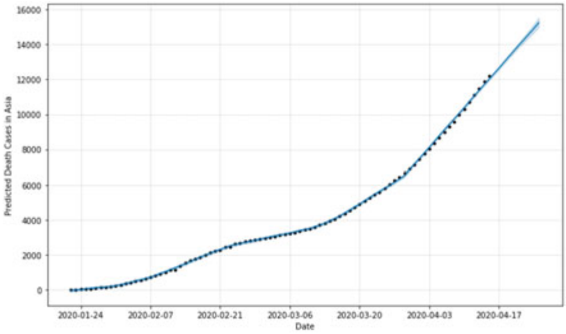


Fig. 35 Predicted death cases in East Asia y and yhat

Appendix 11

See Fig. 36.

Appendix 12

See Fig. 37.

Fig. 36 Predicted confirmed cases in East Asia

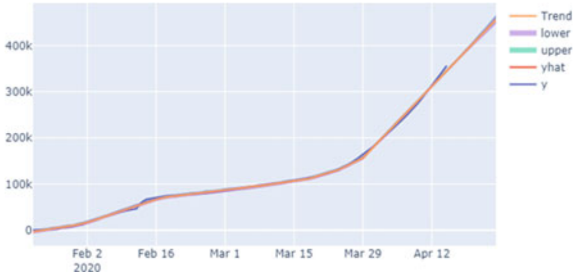
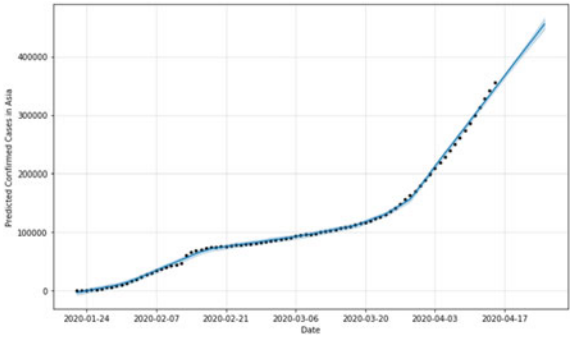


Fig. 37 Predicted confirmed cases in East Asia y and yhat



References

1. Continent. https://en.wikipedia.org/wiki/Continent
2. Asia population. https://en.wikipedia.org/wiki/Asia
3. Cradle of civilization. https://en.wikipedia.org/wiki/Cradle_of_civilization
4. World Economic Outlook (October 2018) GDP, current prices. www.imf.org
5. Largest_Economies_in_Asia. Aneki.com. Retrieved 9 Nov 2017
6. Geography of Asia. https://en.wikipedia.org/wiki/Geography_of_Asia
7. Western Asia. https://en.wikipedia.org/wiki/Western_Asia
8. Report for selected countries and subjects. Imf.org. 20 Sept 2017. Retrieved 22 Jan 2017

Protecting Alexa from Remote Voice Command and Inaudible Voice Command Attacks

Pooja Ahalpara and Bhavesh Borisaniya

Abstract Alexa (voice assistant) is an IoT device that provides various services based on user's voice command. It provides various services from providing information about weather and time, can play music, make a phone call, send messages, order online to control other IoT devices. Along with large number of services, it is vulnerable to different attacks such as home burglary, fake orders, invocation confusion, DoS attacks, voice squatting, inaudible voice command attack, remote voice command attacks, etc. These attacks are feasible mainly because of weak or no voice authentication mechanism for commands fired to the voice assistant device. Specifically, the device cannot identify the source of command, i.e. from where the command came and who gave it. Also, this means that the attacker can use recorded voice or a voice inaudible to human to fire any command and Alexa will execute it without any hesitation. In this paper, we proposed an approach that works based on challenge response mechanism. These challenges are kept dynamic, so the attacker cannot have response for the given challenge in hand during attack. Thus, this approach contributes an efficient solution that can protect voice assistant from remote voice commands and inaudible voice commands.

Keywords Voice assistant · Alexa · Remote voice commands · Inaudible voice commands

1 Introduction

Voice assistant devices are IoT devices that are connected to the Internet and provides services based on user's command. There are various voice assistants available in market such as Amazon Alexa, Siri, Cortana, Google Assistant, Bixby, etc.

Amazon lunched Echo and Alexa device, generally known as Alexa in November 2014. Alexa provides many features such as, playing music, playing game with

P. Ahalpara (✉) · B. Borisaniya
Shantilal Shah Engineering College, Bhavnagar, India
e-mail: poojaahalpara25@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_35

Alexa, telling Jokes, manage calendar and emails, manage shopping list, make a phone call, drop a phone call, make any announcement, play news, give information about whether, sports and traffic [1]. It can also control other smart devices in home such as smart bulbs, TV, etc.

Alexa also have access to private information [2]. Main risk factors threatening to the user's privacy and confidentiality are, it stores voice commands and recordings [3], can control various IoT devices with unauthorized commands, it always listen to the environment, lack of input validation and can understand voice command that are inaudible to human being and/or are recorded previously and fired from distance.

In this research work, we focused on two types of attacks *inaudible voice command attacks* and *remote voice command attacks*. As Alexa cannot identify that whether the command is from the human being or recorded and also cannot identify that the command is having frequency of voice that the normal human being voice have, we created a simple challenge response mechanism that will be useful to avoid such attacks. The mechanism will use dynamic challenges that the user will need to respond. As the challenges are dynamic, the attacker will not be able to predict them before attacking through the recorded or inaudible voice commands.

The rest of the paper is organized as follows. We discuss the different types of attacks on Alexa in Sect. 2. Section 3 provides the details of related work. We describe our proposed work in Sect. 4. Results of experiments conducted are provided in Sect. 5. We conclude our work in Sect. 6 with references at the end.

2 Attacks on Alexa

Alexa always listen to the surrounding environment and has access on users' private information. This gives an attacker a platform to fire various command(s) by leveraging its functionality. Following are the attacks that are possible on voice assistant(s). In this section, we provide details about what kind of attacks are possible on Alexa and later in related work section, we discuss different defense mechanisms proposed in the literature for detection/prevention of such attacks.

Invocation Confusion Attack: Alexa runs specific skill on specific command. The attacker can develop an attacking skill with the invocation name which is similar to other skill or word frequently spoken by the user [4]. The skill can be designed such that it become hard or indistinguishable specially in case of noise. According to Zhang et al. [4], 60% users of Alexa used a word "Please" when lunching a skill. The attacker can create a skill having similar invocation name and that skill invoke some malicious code.

Phishing Attack: A running skill can include a home card in its response to the user. It can be viewed by user through Alexa mobile application or Web browser. For users, it is hard to remember the information and activity stored in history. Attacker can take an advantage of it. Attacker can send for example, notification of account expiration or a reward winning malicious link for further process. The click of user on that link can cause disclosure of him/her credentials [4].

Kumar et al. [5] designed an attack targeting the American Express skill. These skills allows to do banking tasks. For example, "Alexa, ask Amex to make a bank transfer of 3 dollars". As the skill contain a compulsory word "Amex" attacker can publish a skill "Am X". Alexa might execute skill with invocation word "Am X", while user suppose that the genuine skill is invoked. The malicious skill invocation result into a prompt having similar visual as of genuine Amex login page and user will enter his/her credentials and hand it to the attacker thinking that they are giving it to genuine authority and this could lead into financial loss or illegal circumstances.

Remote Voice Command Attack: Yuan et al. [6] proposed REmotE VoicE (REEVE) control attack that remotely command the Alexa by broadcasting the signals through compromised TV, radio, wireless speakers, etc. Attacker can include the attack payload with the signal of radio, inject the signal to specific TV channel to replace original TV channel or play audio over the air using speaker. However, it requires the attacker to know, which channel the radio is operating, user's favorite channel or pass-code of speaker.

In Communication Skill Switch Attack: According to survey by Zhang et al. [4], Alexa supports skill switching during the interaction. The running skill can hand over the control to target skill in a response to the switch command. The sensitive information of user might get shared with the attack skill that can lead to privacy leakage and financial loss.

Hijacking Built-in-Alexa Command: For users, the built-in-Alexa commands are trustworthy. However, the attacker can hijack the built in Alexa command and then redirect it to the malicious skill by associating some utterance with an intent of malicious skill. Hence, Amazon Voice Service (AVS) invoke the skill using an intent corresponding to that utterance. The malicious skill can then return any response to AVS which is then given to the user [4]. This is very dangerous because, the user unintentionally invokes the malicious skill with the belief that the built in command will invoke the right skill only.

Skill Redirection Attack/Jamming User command: Jamming the user command prevents Alexa from understanding the genuine invocation of skill. According to Mitev et al. [7], this attack can be performed by playing inaudible noise every time while user is speaking the command after the wake up word. After the jamming of user command, attacker can invoke the malicious skill and make AVS to start session with it without the knowledge of the user.

Skill-in-the-Middle Attack: This attack gives full power to the attacker to control the conversation of victim and Alexa functionality in specific skills. For example, attacker performs skill jamming after that invoke malicious skill and then can start sessions with AVS, one for malicious skill and other for genuine skill that user wanted to invoke. Attacker can obtain data from genuine skill and can modify it or use it for any other purposes. The modified data can be then given as response to user through malicious skill [7]. This can mislead the user by giving false information.

Inaudible Voice Command Attack: Ultrasound frequency(UF) is inaudible to human beings. The microphone can capture and understand the UF. Attacker use this capability of microphone to attack on Alexa.

Zhang et al. [8] generated the Dolphin Attack. It uses inaudible voice injection. Amplitude modulation technique is used to modulate the audible voice signal to inaudible voice signal. The modulation is performed such that it can be efficiently demodulated by VCSs. The attack can be performed on Alexa, Siri, Google Now, etc. Zhang et al. [9] developed Vaspy that uses AI and target on the VA in Android phones. Vaspy choose the optimal time of attack and the volume of voice. It launches the attack without getting noticed by the users. It can leak private information, call any number, send emails, etc.

DoS Attack: DoS attack make device unavailable for the legitimate users by flooding large number of request to the device. Overstreet et al. [10] used syn-flood in Kali Linux to repeatedly sense synchronize request to the voice assistant. This results in the crash of the system and disconnect from the network. Huraj et al. [11] developed a UDP-based Distributed Reflective Denial of Service (DRDoS) attack which floods the packet to the reflector device. The source IP address is set to the victim's IP address who obtain the reflected replies. Hence, the victim is indirectly overloaded.

Faking Termination Attack: Alexa allows the skill to terminate after the voice response. This termination can be indicated by a small beep or silent audio. Attacker can attack on the skill and make it to pretend the termination by small beep or silent audio while the session is on in background. This is possible because the Alexa let skill to include reprompt when it does not get any other command from the user within specific period of time. Zhang et al. [4] created an attack skill that include a silent audio file up to 90 s in its reprompt. Because of this, it can run for minimum 120 s on Alexa. The running time can be further extended by attaching the silent audio after its last voice response.

3 Related Work

There exist some notable defense mechanisms which are aimed to provide security to voice assistant(s).

Defense Mechanisms Against Remote Voice Command (RVC): Lei et al. [12] designed a VSButton that provides physical presence-based access control mechanism. It detects the physical presence with the help of Wi-Fi technology. If the VSButton is not in push state, then the device will not consider any command. VSButton will be in push state only when there is a physical presence. It will prevent the Alexa from executing the commands given when there is no physical presence, and hence, it provides the protection to Alexa and user from any command given in absence of user. However, the system cannot distinguish the presence of human and other things or lives. Hence, attacker can get into area that make the VSButton in push state or any other living thing like movement of pet can also cause the VSButton to go in push state.

Yuan et al. [6] proposed two factor authentication to provide defense against REEVE attack where Alexa acts as a chatbot whenever it encounter voice command critical to security. It asks some questions based on user's historical profile registered previously, and user is required to give right answer directly through voice.

Huan et al. [13] proposed a wearable solution to provide voice authentication. The wearable device can be placed in ear-buds/locket/neckless/handset/earphone /eyeglasses. It collects the vibration of body surface at where it is placed (either eye/ear/neck backside) and continuously matches that vibrations with voice command received by the voice assistant. It requires that the wearable device touch the body surface and will not authenticate the genuine user if it does not find the vibration from body surface. Also, languages may have some words that does not generate much vibration that can bother the user, as in such case, the command will not be executed.

Chen et al. [14] proposed a way that differentiate machine based voice based on magnetic field which comes from loud speakers and human voice. However, approach suffers from high false positive rate when other devices generating magnetic signals are nearby.

Defense Mechanism Against DoS Attack: Paudel et al. [15] created a graph-based outlier detection in IoT device network that can detect the DoS attack in real time. Using this, it can be identified that on which part of graph there exist a more than usual traffic hence on which device there exist more than usual traffic.

Defense Mechanism Against Voice Masquerading Attack: Zhang et al. [4] created defense system against voice masquerading attack that include two components - Skill Response Checker (SRC) that checks the responses from malicious skill if found suspicious and User Intention Classifier (UIC) that checks the voice commands issued by user to check whether it attempts to switch to different skill in wrong way. As all the real-world attacks were not considered in dataset, there is a possibility of some attacks may bypass this defense system.

Defense Mechanisms Against Inaudible Voice Commands Attack: Kasmi et al. [16] proposed that when voice command is being processed for smart phones, upon detection of any abnormal electromagnetic activity, the command should be rejected.

Mitev et al. [7] suggested the solution to the attack by changing the hardware microphone such that it suppress the command in ultrasound frequency. Another solution is to place a hardware between amplifier and low pass filter (LPF) that can detect and identify the demodulated voice command. A machine learning module can be used to classify the demodulated and recorded audio. Alexa's coding can be changed such that it does not execute the commands given under a loud voice to prevent the jamming [7].

Zhang et al. [8] used support vector machine (SVM) to classify recorded audios to protect system against Dolphine attack. They also designed a mechanism to place a module before LPF that distinguish between recorded audio and human voice.

Giuseppe et al. [17] developed a system AuDroid that detect low-error rate and give access to apps, system services and device owner to run configurations in Safeway.

4 Proposed Work

Vulnerability to remote voice commands and inaudible voice commands allow attacker to fire command from remote location and/or without the knowledge of the user by releasing commands that are inaudible to human being (using ultrasound frequencies). Therefore, even if the user is nearby the device, he/she cannot know that the device is under attack. These may cause the information, financial or reputation loss to the user. Both these attacks can be prevented by identifying whether the command is recorded audio or given by human being.

General idea behind our proposed mechanism is that, if the command is given by a human being then he/she must be around the area within which Alexa can be accessed and be able to answer dynamic question asked by Alexa. However, if the command is given through recorded audio, then that recorded audio will not be able to answer back to the dynamic question asked by Alexa.

Through this approach, whenever Alexa will encounter the command that is critical to security it will give user a challenge in form of a simple mathematical puzzle. After that, it will wait for the response. Once, response is given, then it will check for the correctness and if it is correct then only it will execute given command. In both other cases of wrong response or no response, it will reject the command.

Figure 1 shows the working of proposed method. As shown in Fig. 1, the Alexa will encounter a command first. After encountering the command, it will invoke skill, namely “Home Secure Service” we developed in backend. Upon invoking the skill, if the command is not critical to security then it will simply perform the command and return the result to user through Alexa. And if the command is critical to security then, it will go for security check. In security check, flag having value 1 indicates the permission to execute the command and flag having value 0 indicates command have no permission to be executed yet.

So, whenever the system encounters the command critical to security, it will be sent for permission granting or denying. Over there, at very first if the flag is 0, then it will provide a puzzle that Alexa will speak and then waits for specific time duration (default timeout) for response. If the response is given to Alexa, it will check the correctness of response. And if the response is correct, then only the permission will be granted for command execution and this will be indicated by changing the flag value to 1. Else if the response is encountered and is incorrect or no response encountered, the command will be denied and this will be indicated by flag value 0.

The reason to ask a puzzle is that, if the command is given by the human being, then he/she can solve and can give answer to simple yet dynamic question. However, if the command is given by the attacker in form of remote voice command and/or in inaudible format using some device, then it will not be able to answer back the

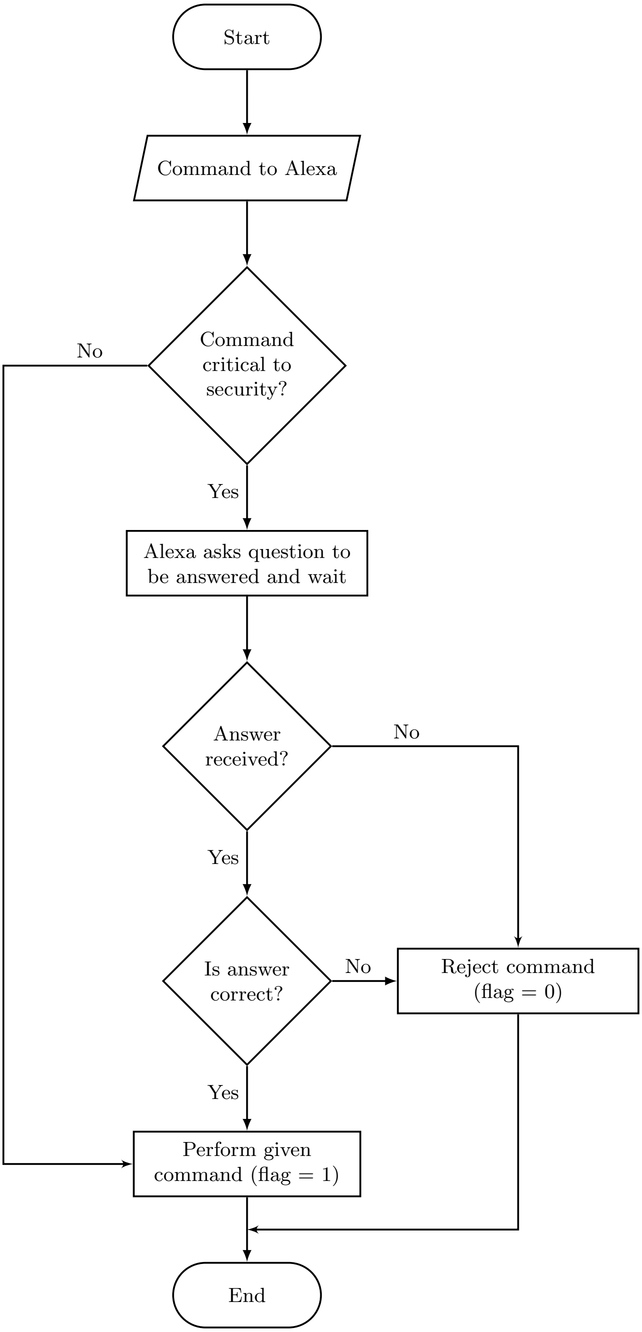

Start
Command to Alexa
Command critical to security?
No
Yes
Alexa asks question to be answered and wait
Answer received?
No
Yes
Is answer correct?
No
Reject command (flag = 0)
Yes
Perform given command (flag = 1)
End

Fig. 1 Flow of proposed work

simple yet dynamic question. This mechanism can easily differentiate the command from human being and from any device in any form.

After returning from the security check, if flag value is still 0, i.e. not changed, the system will send response to Alexa that service is denied. And if flag value is 1, the command will be executed by the system and service/response will be given to the user through Alexa.

5 Experiments and Results

For experiments, we took 74 voice commands given to Alexa to perform particular service. The commands are categorized as critical or non-critical commands as given in Tables 1 and 2. The commands are related to email, phone number, name, address, fun fact services that we considered in our work. Out of 74 commands, 60 commands were critical commands that requires security and 14 were non-critical commands. Below example will give a clear understanding of how the mechanism will work.

For example, if user will ask "what is my name?" or command related to inquiry of user name, then our mechanism will simply return the name without any security check as it is non-critical command. Services such as asking name, weather information, etc., are considered as non-critical service and did not require any security check before providing the response. However, for example, if user will ask "what is my email address?" or command related to the inquiry of the email address of user, then will provide a challenge and expect the response from the user. If the correct response is given, then the command will be accepted and proceed further. Otherwise, in the case of incorrect response or no response, the command or service will be denied.

In our experiments, we tested all 74 commands. Our developed Alexa skill is clearly able to distinguish between critical and non-critical commands. The non-critical commands were executed directly and for critical commands, the security check is performed. Table 3 gives summary of the results that we get after performing the experiments.

As per the results, we can say that our mechanism works well against the test cases we provided. In addition, it eliminates requirements of already existing solutions such as wearing external device [13] or being present in particular area to detect presence [12] and provide a simple yet efficient prevention mechanism against remote voice commands and inaudible voice command attacks.

6 Conclusion

Alexa as a voice assistant provides services just by a user's voice commands. They have wide range of services that make life of human being much easier by providing easy access to various services. Along with these services voice assistants are vulnera-

Table 1 List of critical commands used in experiments

Critical commands	
Will you please send mail	Give my telephonic number
Send mail	What is my personal phone number
Please send mail	What is my address
Send email	Give owner's address info
Email to	Give my address details
Mail to	Where do I live
Compose mail	Where your owner live
Hey can you send mail	What is my postal address
Quickly send mail	What is my residential address
Send electronic mail to	How do I reach home
Send Gmail	What is my Zipcode
Send mail for me	What is my home location
What is my phone number	Where is my home
Give my phone number details	Where is my apartment
Which phone number is registered with you	Where is my house
Give my phone number	Where is my habitat
What's my phone number	Where is my residence
My registered phone number	In which residence I live
My Phone number	What is my Email ID
What is my contact no.	Give details of my email
What is my registered contact number	What's my email
What is my mobile number?	Which mail is registered with you
Phone number	Give my email ID
Which contact number is registered with you	What is my mail
What someone will dial to contact me	What is my electronic mail id
What is my cell phone number	What is my gmail account detail
What is my telephone number	My gmail account ID
What is my directory number	Provide mail Id details
What is my calling number	Tell me my gmail Id
What is my dialing number	What is my gmail account ID

Table 2 List of non-critical commands used in experiments

Non-critical commands	
What's my name	What is the fact of day
Who am I	Fact of the day
What will you call me	Tell me facts
Greet me with my name	Which fact you would like to tell me today
Do you know my name	What is today's fact
Say my good name	What is the fact you will let me know today?
Tell me the fact of the day	Let me know fact of the day

Table 3 Summary of experiment results

Total number of commands	74
Commands critical to security	60
Commands not critical to security	14
Total number of commands recognized	74
Total number of commands recognized which are critical to security	60
Total number of security checks	60
Total number of commands recognized which are not critical to security	14
Total number of direct access	14

ble to many attacks that can impairment user's social image, privacy breach, financial loss, etc. These attacks are mainly feasible due to no or weak input validation in voice assistants. Attackers take advantage of it and can penetrate into system very easily. So, there is a requirement to deploy mechanism(s) that can authenticate the user and validate the input (user's command) before executing it. We proposed a solution to provide security to voice assistant device using challenge response mechanism for skills critical/sensitive to the security. The proposed approach considered a simple and dynamic mathematical puzzle as a challenge and expected a correct response from the user. We have successfully validated our approaches against different test cases.

Moreover, the mechanism we developed is not dependent on any kind of physical presence or movement in some specific area, which does not require any extra hardware to be attached to device or even to human being, does not have any extra cost for installation and does not require any expertise in order to use it. The proposed approach is a simple yet efficient software-based solution that users can utilize to protect their Alexa device from remote voice commands and inaudible voice commands attack.

References

1. Alexa A. Alexa skills. https://www.amazon.com/alexa-skills/
2. Alepis E, Patsakis C (2017) Monkey says, monkey does: security and privacy on voice assistants. IEEE Access 5:17841–17851
3. Javed Y, Sethi S, Jadoun A (2019) Alexa's voice recording behavior: a survey of user understanding and awareness. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3339252.3340330
4. Zhang N, Mi X, Feng X, Wang X, Tian Y, Qian F (2019) Dangerous skills: understanding and mitigating security risks of voice-controlled third-party functions on virtual personal assistant systems. In: 2019 IEEE symposium on security and privacy (SP), pp 1381–1396
5. Kumar D, Paccagnella R, Murley P, Hennenfent E, Mason J, Bates A, Bailey M (2019) Emerging threats in internet of things voice services. IEEE Secur Privacy 17(4):18–24

6. Yuan X, Chen Y, Wang A, Chen K, Zhang S, Huang H, Molloy IM (2018) All your alexa are belong to us: a remote voice control attack against echo. In: 2018 IEEE global communications conference (GLOBECOM), pp 1–6
7. Mitev R, Miettinen M, Sadeghi AR (2019) Alexa lied to me: skill-based man-in-the-middle attacks on virtual assistants. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3321705.3329842
8. Zhang G, Yan C, Ji X, Zhang T, Zhang T, Xu W (2017) Dolphinattack: inaudible voice commands. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3133956.3134052
9. Zhang R, Chen X, Wen S, Zheng X, Ding Y (2019) Using AI to attack VA: a stealthy spyware against voice assistances in smart phones. IEEE Access 7:153542–153554
10. Overstreet D, Wimmer H, Haddad RJ (2019) Penetration testing of the amazon echo digital voice assistant using a denial-of-service attack. In: 2019 SoutheastCon, pp 1–6
11. Huraj L, Simon M, Horák T (2018) IoT measuring of udp-based distributed reflective dos attack. In: 2018 IEEE 16th international symposium on intelligent systems and informatics (SISY), pp 000209–000214
12. Lei X, Tu GH, Liu AX, Li CY, Xie T (2018) The insecurity of home digital voice assistants—vulnerabilities, attacks and countermeasures. In: 2018 IEEE conference on communications and network security (CNS), pp 1–9
13. Feng H, Fawaz K, Shin KG (2018) Wearable technology brings security to alexa and siri. GetMobile Mobile Comput Commun 22(1):35–38. https://doi.org/10.1145/3229316.3229328
14. Chen S, Ren K, Piao S, Wang C, Wang Q, Weng J, Su L, Mohaisen A (2017) You can hear but you cannot steal: defending against voice impersonation attacks on smartphones. In: 2017 IEEE 37th international conference on distributed computing systems (ICDCS), pp 183–195
15. Paudel R, Muncy T, Eberle W (2019) Detecting DoS attack in smart home iot devices using a graph-based approach. In: 2019 IEEE international conference on big data (big data), pp 5249–5258
16. Kasmi C, Lopes Esteves J (2015) Iemi threats for information security: Remote command injection on modern smartphones. IEEE Trans Electromagn Compat 57(6):1752–1755
17. Petracca G, Sun Y, Atamli A, Jaeger T (2016) Audroid: preventing attacks on audio channels in mobile devices

An Ontological Model to Support the Application of Quality Practical Guidelines to Assess Business Process Descriptions

Olga Y. Rojas, Nemury Silega, Ashutosh Sharma, Yuri I. Rogozov, and Vyacheslav S. Lapshin

Abstract Business process models are a powerful means to show the business view of an organization. These models enhance the common understanding about the process of the organization and represent a key artifact to design information systems. Therefore, detecting and correcting errors in business process models are crucial to prevent errors in these models from spreading to other stages of the software development process. However, several studies have demonstrated that business process models usually have errors. The use of semiformal notations is one of the reasons for this fact, since these notations make the semantic validation of models difficult. Ontologies has become in a suitable solution to represent business process models. Since ontologies are a formal language based on description logics, its adoption enables semantic validation of modes. This paper aims to describe an ontology-based approach to detect errors in business process models. This approach implements a set of practical guidelines to assess the quality of process models. To develop the ontology, a solid methodology was followed. Likewise, the ontology was validated through a recognized method. Some examples that illustrate the applicability and impact of this approach are provided.

Keywords Business process models · Ontology · Quality practical guidelines

O. Y. Rojas
Universidad de Las Ciencias Informáticas, Habana, Cuba
e-mail: yarisbel@uci.cu

N. Silega (✉) · A. Sharma · Y. I. Rogozov · V. S. Lapshin
Department of System Analysis and Telecommunications, Southern Federal University, 347900 Taganrog, Russia
e-mail: silega@sfedu.ru

A. Sharma
e-mail: ashutosh@sfedu.ru

Y. I. Rogozov
e-mail: yrogozov@sfedu.ru

V. S. Lapshin
e-mail: lapshin@sfedu.ru

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_36

1 Introduction

Business process modeling is a complex task where the knowledge of different people is combined, for example business specialists, stakeholders, and analysts [1]. The tacit knowledge of the business experts is an usual challenge during the business process modeling [2].

For the software development, business process models are a useful artifact for the software developers in order to understand the business of the companies [3–5]. Hence, to obtain free of errors user requirement specifications in the first stages of the information systems development process, the modeling of business processes is essential [6–8]. However, several studies provide evidences that demonstrate the existence of errors in business process model usually [7, 9]. The wrong modeling of business processes and its lack of correspondence with the system design are usual factors for the systems failure [10].

In spite of the relevance of process modeling, often is carried out by inexpert modelers [11]. Therefore, the quality of the models is affected in terms of flexibility and completeness [12, 13]. Even, sometimes modelers are not aware of the importance of their work to prevent that errors in these models from spreading to other stages of the software development process [14, 15].

Since the importance of the business process modeling for the software development, its evaluation is critical for the success of the process. In that sense, the understandability is a key quality indicator of business process models [16]. The understandability of models can be related to structural properties of its graphical elements [6]. Some practical guidelines to evaluate the understandability of business process models have been proposed [15]. However, it is not easy to find proposals that support the automatic application of these practical guidelines.

On the other hand, ontologies have become in a suitable technology to represent, validate, and analyze business processes [17]. The description of business processes in an ontology helps to detect model inconsistencies automatically and avoids the propagation of errors to system models [18].

This paper aims to describe an ontological model to assess business process descriptions. The assessment is based on practical guidelines to check the structural properties and the understandability of models [15]. Some specifications to automatically apply these guidelines in the ontology were implemented. To ensure the quality of the ontology, a solid methodology was adopted. Likewise, a procedure to evaluate the ontology was applied.

The rest of the paper is organized as follows. In Sect. 2, the practical guidelines and the basic technologies to develop the ontology are introduced. In Sect. 3, an ontological model to assess business process descriptions is described. In Sect. 4, some examples to illustrate the applicability of the approach are presented. Section 5 presents the conclusions and future work.

2 Background

Several works that deal with the quality of business process descriptions were analyzed [4, 6, 19–24]. Some authors have proposed practical guidelines to enhance the quality of the business process descriptions [6, 25, 26]. The application of these practical guidelines may reduce errors and improve the understandability of the business process models. However, some practical guidelines are not support by the modeling tools. Therefore, an alternative to support the application of these practical guidelines may be a useful contribution.

Several authors foster the adoption of ontologies to represent and validate business process descriptions. An ontological approach may support the identification of errors and improve activity labels of process models [2, 17, 18]. Therefore, we have adopted an ontological approach to support the application of practical guidelines to assess business process descriptions.

2.1 *Quality Practical Guidelines*

Table 1 gives ten practical guides related to the size of business process models. These guidelines allow to classify business process models according to the number of modeling elements they have [21, 27].

Table 1 Practical guidelines regarding size

Size problem	Guideline
P1. *High number of elements*	More than 31 elements in a model must with be avoided
P2. *High number of events*	More than seven events in a model must be avoided
P3. *High number of start event*	More than two events in a model must be avoided
P4. *Lack of start events*	The start event cannot be missed
P5. *High number of end events*	More than two end events in a model must be avoided
P6. *Lack of* end *events*	The end event cannot be missed
P7. *High number of intermediate events*	More than five intermediate events in a model must be avoided
P8. *High number of sequence flows*	More than 34 sequence flows in a model must be avoided
P9. *High number of* gateways	More than 12 gateways in a model must be avoided
P10. *High number of activities*	More than 31 activities in a model must be avoided

2.2 *Methodology and Tools to Support the Development of the Ontology*

The selection of a proper methodology is a key step to develop an ontology. To guide the development of the ontology presented in this paper, the methodology of Noy and McGuinness was adopted [28]. This is a solid methodology that has been widely adopted and includes the following steps:

- Determine the domain and scope of the ontology.
- Consider reusing existing ontologies.
- Enumerate important terms in the ontology.
- Define the classes and the class hierarchy.
- Define the properties (called relationships or slots) of the classes.
- Define facets and/or restrictions on slots or relationships.
- Define instances.

Web Ontology Language (OWL) [Ref] was adopted to represent the ontology. OWL is based on description logics and includes the operator's intersection, union, and negation which are very useful to represent knowledge. Furthermore, the models represented in OWL can be analyzed by reasoners which automatically check the consistency and infer new knowledge. The reasoner Pellet was adopted to analyze our ontology. To implement the ontology, the tool Protégé [Ref] was adopted. Protégé is multiplatform and open-source, and it has a flexible and extensible architecture. The language OWL and the reasoner Pellet are supported by Protégé.

3 An Ontological Model to Support the Application of Practical Guidelines to Assess Business Process Description

To develop the ontology, the steps defined in the methodology of Noy and McGuinsess were carried out. Below the main results are described.

Step 1. Determine the domain and scope of the ontology

The ontology has the purpose of assessing the quality of the business process descriptions applying practical guidelines. To achieve this objective, the ontology must be able of answering the following competence questions (CQ):

1. Does the process meet basic workflow patterns?
2. What activities are included in the process?
3. What processes have problem of size?
4. What processes have problem of morphology?
5. What processes are efficient?
6. What processes are inefficient?

7. What processes are very efficient?
8. What processes are very inefficient?
9. What processes are low efficient?

Step 2. Consider reusing existing ontologies

Concepts of the ontology described by Silega and Noguera were reused [18]. This ontology supports the description of business processes and includes specifications for its validation.

Step 3. Enumerate important terms in the ontology

The main terms are related with the components of a process such as activity, event, gateway, input, output, and others terms. Furthermore, terms related with the application of practical guidelines (see Table 1) to assess the models are considered.

Step 4. Define the classes and the class hierarchy

To define the classes of the ontology, three main elements were considered. First of all, we considered the concepts related to the representation of business process, for this regard, the concepts of the ontology developed by Silega [18], such as **Processs**, **Activity**, **Event,** and **Gateway** were reused. Likewise, this ontology includes classes such as **Step** and **FlowElement** to represent the flow of activities. Furthermore, we included classes to assess the processes based on the practical guide. For example, to classify the processes that do not fulfill some practical guideline related to the size, the class **ProcessWithSizeProblem** was included. The classes ProcessWithMorfologyProblem, **ProcessEfficient**, **ProcessInefficient**, **ProcessVeryInefficient**, and **GatewayWithProblem** are related to the application of the practical guidelines too. These last classes were declared as defined classes. Defined classes in OWL include a set of necessary and sufficient conditions, thus a reasoner can automatically infer their instances. Hence, the process with problems will be automatically identified. Figure 1 shows an excerpt of the class hierarchy.

Step 5. Define the properties

Properties are the other core component of ontologies. Object properties and data properties are the two types of properties in an ontology. The object properties allow to represent a binary relationship between two individuals. Each object property has an inverse object property, for example, an **Activity** *belongsTo* a **Process** and a **Process** *HasActivity* an **Activity**. A total of 64 object properties in the ontology were defined. Table 2 gives some object properties related to the classes Process and Step.

After creating the properties, it is possible to declare some necessary and sufficient conditions to automatically classify the instances of the defined classes. For example, Fig. 2 shows the necessary and sufficient conditions to identify the processes that belong to the class **ProcessWithSizeProblem**.

Other rules to assess the processes also have been implemented. In spite of the expressivity richness of OWL, some complex relations cannot be expressed.

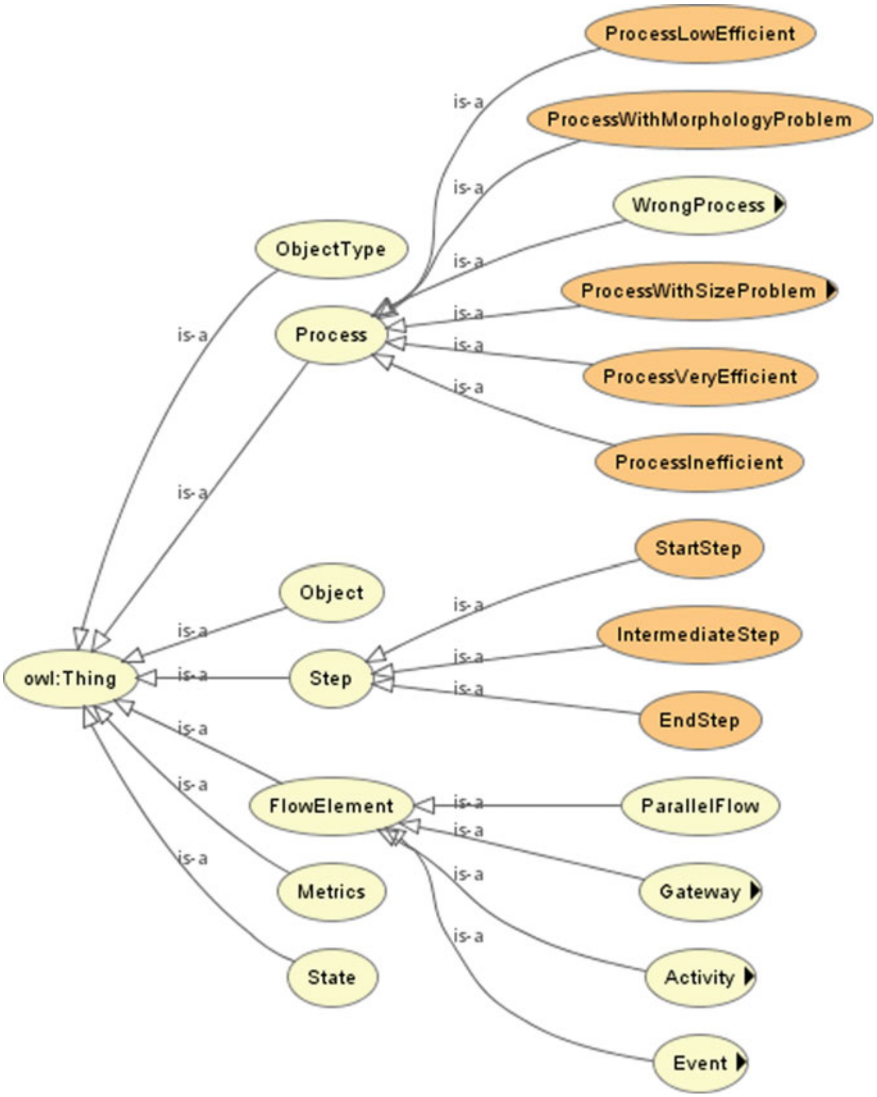


Fig. 1 Classes hierarchy

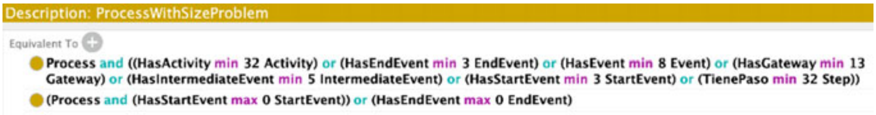
Therefore, the semantic web rule language (WSRL) was adopted to complement OWL.

Step 6 Define instances

An example to illustrate the creation of instances in the next section is presented.

Table 2 Examples of object properties

Domain	Object	Range
Process	***HasActivity***	Activity
Process	***HasStartEvent***	StartEvent
Process	***EndEvent***	StartEvent
Process	***HasStep***	Step
Process	***HasProblemWithMetrics***	Step
Step	***ExecutesTo***	Metrics
Step	***FollowsTo***	FlowElement


Description: ProcessWithSizeProblem

Equivalent To

● Process and ((HasActivity min 32 Activity) or (HasEndEvent min 3 EndEvent) or (HasEvent min 8 Event) or (HasGateway min 13 Gateway) or (HasIntermediateEvent min 5 IntermediateEvent) or (HasStartEvent min 3 StartEvent) or (TienePaso min 32 Step))

● (Process and (HasStartEvent max 0 StartEvent)) or (HasEndEvent max 0 EndEvent)

Fig. 2 Example of a set of necessary and sufficient conditions

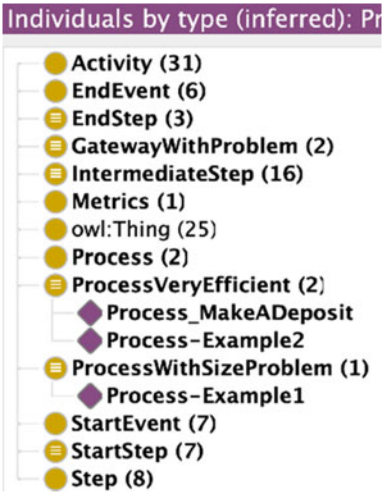
4 Evaluation of the Ontology

Checking that the ontology fulfills its conditions as a logical-formal system is the first step to evaluate its quality. The reasoner Pellet confirmed that our ontology fulfills its conditions as a logical-formal system.

On the other hand, to demonstrate the applicability of our approach, some business processes in the ontology were described and assessed. The processes *Process_Make-A_Deposit* and *Process-Example2* were modeled in the ontology and classified as a **ProcessVeryEfficient** while the process *Process-Example1* was classified as **ProcessWithSizeProblem** because meets the necessary and sufficient conditions of this class. Figure 3 depicts a view of Protégé where the classifications carried out by the reasoner are displayed. This view shows the classifications for *Process_Make-A_Deposit*, *Process-Example1,* and *Process-Example2*. This examples answer the competence questions 4 and 7.

By means of this ontology, other practical guidelines can be verified. For example, it is possible to identify the gateways with multiple input and output flows. A process with this type of gateways should be classified as a process with problem of morphology. Figure 4 depicts an example of a process description with this problem.

To identify the gateways with multiple input and output flows, some rules were specified. The gateways with this problem will be classified as **GatewayWithProblem**. Furthermore, we defined that if a processes has some **GatewayWithProblem** then it is classified as **ProcessWithMorphologyProblem**. After modeling in the ontology, the process of Fig. 4, the reasoner classified *Gateway2* as a **GatewayWithProblem** (Fig. 5a). Since that Process-Example3 has to *Gateway2,* it is classified as a **ProcessWithMorfologyProblem** (Fig. 5b).



Individuals by type (inferred): Pr

- Activity (31)
- EndEvent (6)
- EndStep (3)
- GatewayWithProblem (2)
- IntermediateStep (16)
- Metrics (1)
- owl:Thing (25)
- Process (2)
- ProcessVeryEfficient (2)
  - Process_MakeADeposit
  - Process-Example2
- ProcessWithSizeProblem (1)
  - Process-Example1
- StartEvent (7)
- StartStep (7)
- Step (8)

Fig. 3 Classification of business processes

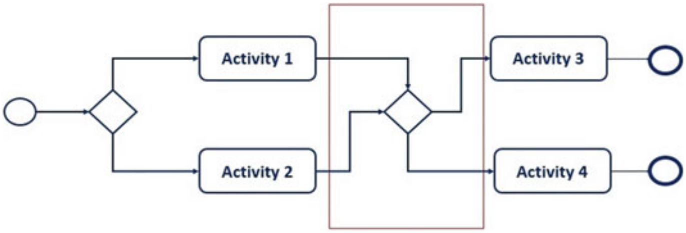
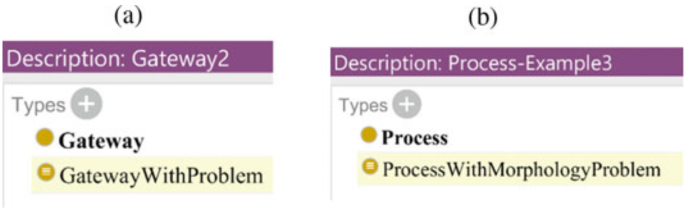


Fig. 4 Model of a process with problem of morphology



(a)

Description: Gateway2

Types

- Gateway
- GatewayWithProblem

(b)

Description: Process-Example3

Types

- Process
- ProcessWithMorphologyProblem

Fig. 5 Classification of a process with problem of morphology

5 Conclusions

The business process models are a useful instrument to understand the business of organizations. Likewise, it is a key artifact to design information systems. Therefore, assuring the quality of process descriptions is crucial to prevent errors in other stages of the software development process. Some quality practical guidelines to evaluate business process descriptions have been proposed. In this article, an ontological model to represent and assess business processes was introduced. The formalization of the process models through the OWL language allows verifying the problems related to non-compliance with the quality practical guidelines related to general complexity. The compliance of these practical guidelines improves the understanding between business experts, analysts, and the development team. The conditions of the ontology as a logical-formal by means of a reasoner were verified. Some examples to illustrate how the expressivity richness of OWL was exploited to represent and assess business process were presented.

Acknowledgements This research is supported by a postdoc fellowship granted by the Institute of Computer Technologies and Information Security, Southern Federal University, project № PD/20-02-KT.

References

1. Marlon Dumas MLR, Mendling J, Reijers HA (2013) Fundamentals of business process management. Springer, Austria
2. Júnior VHG (2016) Utilização de Ontologias para Certificação de Boas Práticas em Modelagem de Processos de Negócio. Universidad de Federal Do Rio Grande Do Sul, Brasil, pp 13–14
3. Jean Carlos Guzmán FL, Matteo A (2013) Del Modelo de Negocio a la Arquitectura del Sistema considerando Metas, Aspectos y Estándares de Calidad, in Revista Antioqueña de las Ciencias Computacionales y la Ingeniería de Software, RACCIS, Venezuela, pp 19–37
4. Isel Moreno Montes de Oca MS, Reijersc HA, Morffi AR (2015) A systematic literature review of studies on business process modeling quality. Inform Softw Technol 58:187–205
5. Méndez R, Urrutia A (2016) Complejidad en modelos conceptuales de procesos de negocios. Propuesta de métricas de calidad de modelos conceptuales de procesos. Revista GTI 15(43):47–62
6. Mendling JHAR, van der Aalst Barjis WMPJ (2010) Seven process modeling guidelines (7PMG). Inform Softw Technol 52:127–136
7. Silega N (2014) Método para la transformación automatizada de modelos de procesos de negocio a modelos de componentes para Sistemas de Gestión Empresarial, in Centro de Informatización de Entidades. Universidad de las Ciencias Informáticas, La Habana, Cuba, p 5
8. Ahmet Dikici O, Demirors O (2018) Factors influencing the understandability of process models: a systematic literature review. Inform Softw Technol 93:112–129
9. Mendling J (2009) Empirical studies in process model verification, vol 5460. Springer, Berlin, Heidelberg, pp 208–224
10. Barjis J (2008) The importance of business process modeling in software systems design. In: Science of computer programming, 2008. Science of Computer Programming, pp 73–87
11. Recker J et al (2009) Business process modeling-a comparative analysis. J Assoc Inf Syst 10(4):1
12. Leung FBN (2005) Analyzing the quality of domain models developed by novice systems analysts. In: Proceedings of the 38th annual Hawaii international conference on system sciences. 2005, IEEE Computer Society, pp 188.2
13. Samira Si-Said SA, Comyn I (2013) Improving business process model quality using domain ontologies. J Data Semant 2:75–87

14. Mendling J, et al (2006) A quantitative analysis of faulty EPCs in the SAP reference model. BPM Center Report BPM-06-08, BPMCenter.org
15. Oca IMMD (2015) Patrón y clasificación taxonómica para directrices prácticas en modelos de procesos de negocio. In: *Departamento de Computación*, Universidad Central "Marta Abreu" De Las Villas, Santa Clara, Cuba
16. James Nelson GP, Genero M, Piattini M (2012) A conceptual modeling quality framework. Software Qual J 201–228
17. Gassen JB et al (2017) An experiment on an ontology-based support approach for process modeling. Inf Softw Technol 83:94–115
18. Silega N, Noguera M (2021) Applying an MDA-based approach for enhancing the validation of business process models. Procedia Comp Sci 184:761–766
19. Aguilar-Saven RS (2004) Business process modelling: review and framework. Int J Prod Econ 90(2):129–149
20. Becker J, Rosemann M, Von Uthmann C (2000) Guidelines of business process modeling. Business process management. Springer, pp 30–49
21. Flavio Corradini AF, Fornari F, Gnesi S, Polini A, Re B, Spagnolo GO (2017) Quality assessment strategy: applying business process modelling understandability guidelines. Italia
22. Figl K (2017) Comprehension of procedural visual business process models. Bus Inf Syst Eng 59(1):41–67
23. Genero M et al (2011) Research review: a systematic literature review on the quality of UML models. J Database Manage (JDM) 22(3):46–70
24. O'Neill P, Sohal AS (1999) Business process reengineering a review of recent literature. Technovation 19(9):571–581
25. Moreno-Montes de Oca I, Snoeck M (2014) Pragmatic guidelines for business process modeling. T.R. 2592983, Editor. 2014, Faculty of Economics and Business, KU Leuven
26. Corradini F et al (2017) A Guidelines framework for understandable BPMN models. Data Knowl Eng 113:129–154
27. Sánchez-González L, et al (2010) Quality assessment of business process models based on thresholds. In: OTM confederated international conferences on the move to meaningful internet systems. Springer
28. Noy NF, McGuinness DL (2001) Ontology development 101: a guide to creating your first ontology. Stanford Medical Informatics, Stanford

Quiz Maker: Automatic Quiz Generation from Text Using NLP

Ebrahim Gabajiwala, Priyav Mehta, Ritik Singh, and Reeta Koshy

Abstract In the past few years, there has been great technological advancement in the field of deep learning and natural language processing. One of the applications is automatic generation of quizzes from text. The recent advancement in NLP techniques has shown a lot of promise. The proposed solution uses an NLP pipeline involving Bert and T5 transformers to extract keywords and gain insights from the text input. From the extracted keywords, different types of questions are generated such as fill in the blanks, true or false, Wh-type and multiple choice questions. Latest state-of-the-art models proved to perform better in all stages of our pipeline. The results from these models have shown a lot of promise. Through a survey created for evaluating the model, around 60% questions generated by the model were incorrectly identified as human generated or could not be determined by the survey participants.

Keywords Deep learning · Natural language processing · NLP pipeline · Quiz generation · State-of-the-art

1 Introduction

In this period of Covid pandemic, the major amount of learning in the education sector is being carried out in the online mode. Students are finding numerous distractions and are unable to concentrate on the concepts taught by their teachers. Without the

E. Gabajiwala · P. Mehta (✉) · R. Singh · R. Koshy
Bharatiya Vidya Bhavans Sardar Patel Institute of Technology Munshi Nagar, Andheri (West), Mumbai 400058, India
e-mail: priyav.mehta@spit.ac.in

E. Gabajiwala
e-mail: ebrahim.gabajiwala@spit.ac.in

R. Singh
e-mail: ritik.singh@spit.ac.in

R. Koshy
e-mail: reeta_koshy@spit.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*, Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_37

student being physically present in front of the teachers, they are unable to make sure if the student has paid attention and listened to their lectures. Hence, there is a need for teachers to have tools that would allow them to test their students' understanding after lectures. Also, students themselves are finding it difficult to quickly revise their concepts. Whenever students take a test, they learn the material better, thereby improving their chances of passing in the subject [1]. An innovative way to test the understanding of a particular text will always be helpful for students. This will increase user interest in studies and will act as a helper to the teacher to ensure that students understand the concepts well. Our solution is a system that allows users to generate quiz style questions, flashcards and a summary for a given corpus of text. Our system will provide solutions to students but also for teachers who can test students using this system for the concepts or materials taught in lecture.

In the past, a combination of various deep learning and natural language processing architectures and methodologies have been used in multiple attempts at providing a solution for this problem. We have compared these approaches and explored them and understood their drawbacks. We have then used the latest, state-of-the-art algorithms and models to create a solution that generates quizzes comparable to those generated by humans.

In our study, we use state-of-the-art models like GPT2 and transformers like T5 and BERT at different stages of the pipeline. In the current scenario, there is no concrete solution to the automated generation of quizzes.

Rest of the paper is structured as follows: In Sect. 2, we briefly discuss related works. Section 3 describes the proposed methodology and experiments. Section 4 shows result and discussion. Section 5 gives conclusion.

2 Related Works

In this section, multiple methods have been discussed that can be used for automatic generation of quiz type questions. Over the years, various NLP techniques and different deep learning architectures have been adopted for question generation. Recently, there has been advancement in the field of NLP models and various new deep learning architectures, and models have been proposed which are current state of the art in a lot of NLP-based problems.

Singh Bhatia, Arjun, et al. have performed sentence selection from Wikipedia text by referring to existing MCQs and obtained accuracy of 88%. Key identification is done by recognizing patterns that are likely to extract sentences containing a particular type of entities [2].

P. Pabitha et al. have proposed a system that uses Naive Bayes for supervised learning. It also uses text summarization technique and noun filtering [3].

Aithal, S. G. et al. have generated questions and their answers from the input passage and calculated cosine similarity between generated questions and the input question. Using the similarity score, the input question is classified as answerable (the answer is then given), unanswerable or irrelevant [4].

Srivastava, Animesh, et al. have proposed a system that consists of a state-of-the-art solution that uses a NLP pipeline and image captioning. ResNet 152 is used as an image feature extractor. Stanford CoreNLP is used for dependency parsing. Uses Glove to create a vector representation of words and distractor generation [5].

Liu, Zhuang, et al. showed that reinforcement learning can be used to take semantics along with syntactic relationships into account, for generating a question given a passage and answer as input [6].

Rohde, Tobias, et al. used a hierarchical attention transformer sequence to sequence tasks such as text summarization and machine translation. The model consists of word-level as well as sentence-level attention mechanisms [7].

Wang, Zichao, et al. used QG-net as an improvement over the previous LSTM + attention models for question generation. Bi-LSTM is used as a context reader. The model is trained and tested on the SQuAD dataset. Rouge L score of 0.4437 is obtained [8].

Savelieva, Alexandra, et al. used the BERTSum model for summarizing human speech. Best results were obtained by training the model on data in ordered fashion, i.e., structured data such as news reports and instructional texts first and then data with more complicated language such as video transcripts [9].

Literature survey shows that this problem has been tried with many different NLP and deep learning-based approaches. We also observed that many papers used traditional methods along with NLP and deep learning approaches which was a reason for not getting very good results. We also saw some different approaches like reinforcement learning and QG-net which gave good results. Most of the papers did not mention a proper pipeline for their approaches. Also, we observed that most of the papers did not use latest state-of-the-art models in their approaches or pipelines which could have given promising results. Also, none of the papers were able to generate all types of quiz questions using a single pipeline.

3 Materials and Methods

3.1 Overview

As shown in Fig. 1, the system accepts input in the form of text, audio input or images. The audio or image input will be converted to text form using speech-to-text converter and OCR, respectively. For OCR, we are using the Microsoft Azure Read API, which is capable of identifying text in images containing either printed or handwritten text. For the speech-to-text feature too, we are using the API provided by Microsoft Azure. Now, the text preprocessing module will remove noisy text and stopwords and apply more text preprocessing methods if required. This processed text will then be fed to the keyword extraction module and text summarization module. Further, keyword extraction will be performed on the summarized text. Now, common words from the output of the two keyword extractions will be used for forming questions and

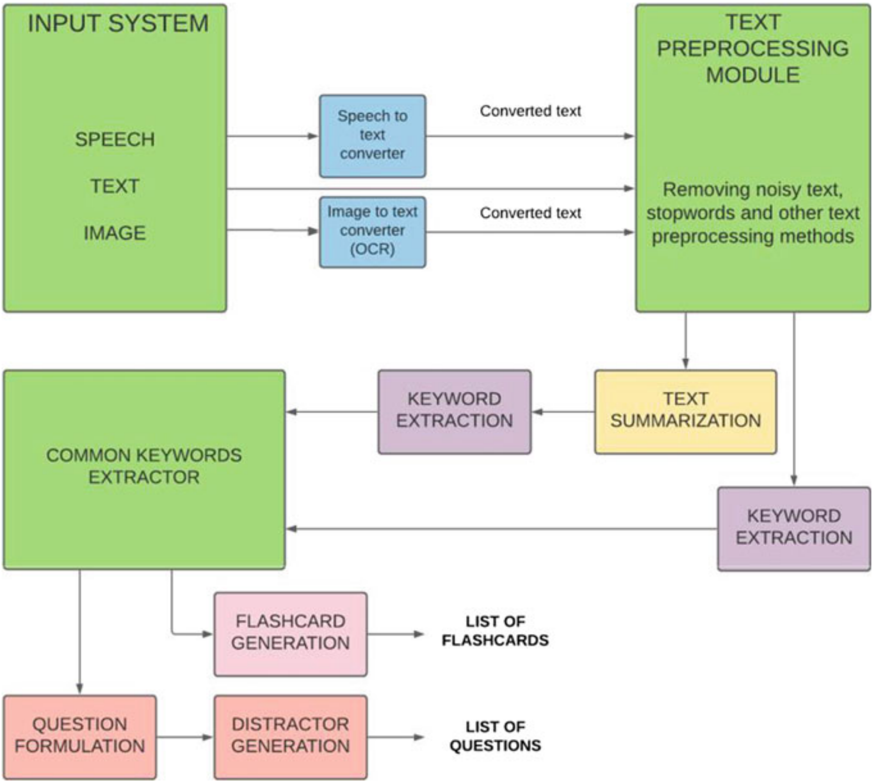


Fig. 1 Methodology block diagram

generating flashcards. Fill in the blanks, true or false and MCQ type questions will be generated. Wh-type questions will be formed, and distractors will be generated for the MCQ questions. These various modules are explained in detail in the further section.

3.2 Models Used in Our Pipeline

BERT the model sentence transformers distilbert base-nli-stsb-mean-tokens is a model implemented in the Transformer library. It is used for natural language processing and is used to compute a 768-dimensional dense vector space for various corpuses, sentences or paragraphs. They can even compute vectors for short keywords or phrases. These models perform state-of-the-art results in various tasks using DistilBERT. These embeddings can then be used for various tasks like clustering, semantic search and sentence similarity by using cosine similarity on these vector spaces. This model proves to be efficient in terms of sentence- or word-level similarities [10, 11].

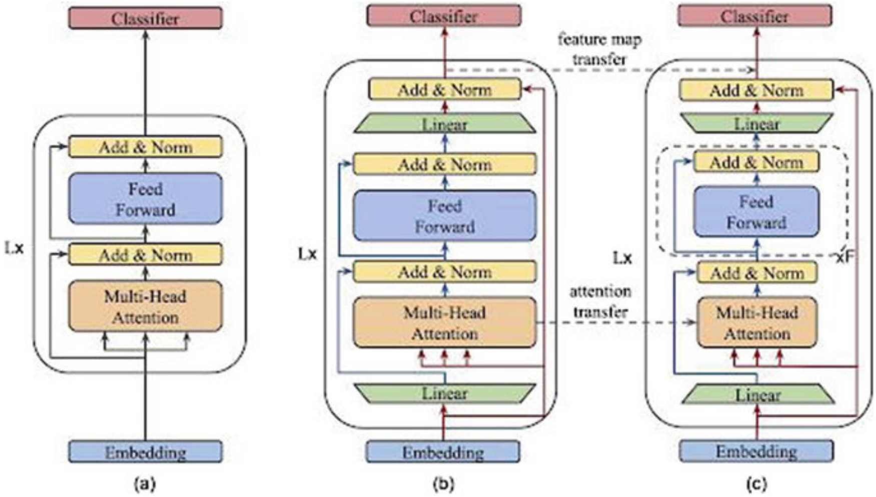


Fig. 2 BERT block diagram

GPT2 is a model from OpenAI, and it is a pre-trained language model that is trained over 40 GB of data. It can be used for various NLP tasks such as translation, summarization and text generation. The architecture of GPT2 is based on the concept of transformers. GPT2 has significantly outperformed other models in the area of text generation for small input content. The output is generated by taking the previously generated data as the input.

T5 transformer was trained on the open-source C4 pre-training dataset. This model can be used on various text-to-text NLP tasks such as question answering, text summarization, sentiment analysis and machine translation. The transformer model has the usual encoder–decoder structure. The encoder and decoder each consist of 12 blocks where each block comprises self-attention, optional encoder–decoder attention and a feed-forward network. In total, the model has roughly 220 million parameters. The model has achieved state-of-the-art performance on various NLP benchmarks such as SQuAD and GLUE. The model is flexible and can be easily trained to perform tasks other than those mentioned by the authors, requiring very few training samples [12] (Figs. 2 and 3).

3.3 Modules in Our Pipeline

Keyword extraction is performed using the BERT transformer. Sentences are converted to BERT embeddings using the DistilBERT embedding model. The parameters such as top n, n gram range and stopwords have been set to 10, 3 and English, respectively. First we extracted candidate keywords, i.e., potential keywords that can

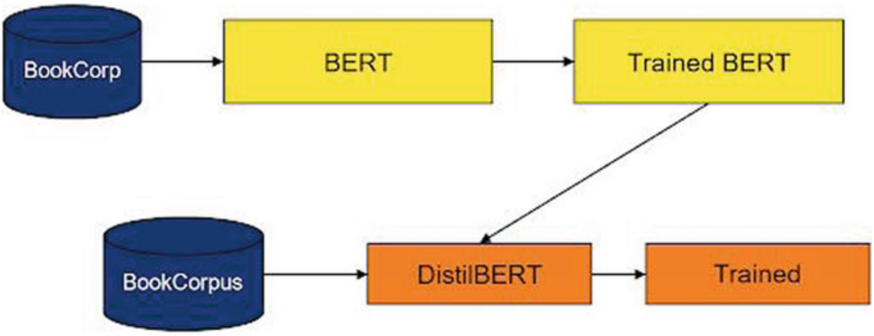
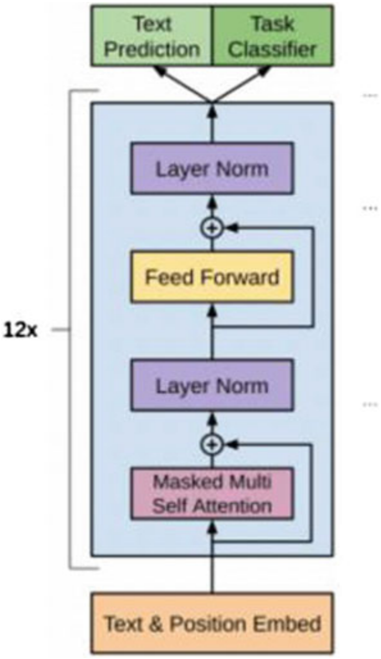


Fig. 3 DistilBERT block diagram

be used to describe the doc. We then used the embedding model to find the embeddings of the candidate keywords and the doc. Cosine similarity is used to calculate the distance between the doc embeddings and the candidate embeddings. Then, the candidate embeddings with the smallest distance are considered for further stages in the pipeline (Fig. 4).

For text summarization, we are using the BERT extractive summarizer. This model has given promising results in the selection of key phrases or key sentences which

Fig. 4 Block diagram of GPT2



represent the input text well. The model uses BERT for text embeddings and K-means clustering for the identification of sentences closest to the centroid, for being included in the summary.

As mentioned earlier, further keyword extraction will be performed on the summarized text and common words from the output of the two keyword ex-tractions will be used for forming questions and generating flashcards. Keyword extraction has been performed twice in order to ensure sentence selection. Sentence selection is basically done to determine which sentences are t enough to create a question for, as it is not possible to create a question for each sentence. It is performed keeping in mind the sentences which attribute the most to the semantic of the corpus are selected.

Fill in the blanks, MCQ type and true or false questions will be generated. Wh-type questions will be formed, and distractors will be generated for the MCQ questions.

For generating fill in the blank type questions, we have used the common keywords and the sentences where they are used. To find the sentences containing the keyword, we have tokenized sentences and mapped them to the keyword they contain. Then, we have simply substituted the keyword with a blank in the sentence.

For generating the multiple choice type questions, we have first used the approach explained above for generating fill in the blanks. Then, we have generated distractors (wrong choices) for the blank. Distractors are basically to generate different alternatives against the correct answer for a given question to confuse the quiz taker. We have used ConceptNet and WordNet for getting distractors which performed really well for keywords with single n gram, i.e., single word keyword. Sense2Vec model is used to generate distractors for keywords whose word length is greater than one. So with these three models, we were able to generate very good distractors for all types of keywords.

For T/F Questions generation, we have used the benepar parser which ex-tracts the verb or noun phrases from the sentence (particularly from the end of the sentence). Post that we feed the remaining sentence to a distill GPT-2 language transformer model which generates complete sentences from the half sentence that we had fed. We took the top three sentences generated from the model and chose the one with the highest dissimilarity score with the actual sentence. We used cosine similarity to find similarity between sentences. We had tried two approaches, one with GPT-2 and another with distill GPT-2 and the latter proved to be much faster than the former.

We have used the T5 transformer for forming Wh-type questions. A sentence and a key phrase or keyword occurring in that sentence are provided as input to the transformer. The transformer uses the sentence as context and then returns a Wh-question whose answer will be the input key phrase or keyword.

For flashcard generation, we extracted the noun phrases from the text and used a Python module named PyDictionary for getting synonyms and meanings for those phrases or sentences. An example for the same can be seen in Fig. 8 wherein a noun phrase, “data”, was extracted, and its flashcard was generated.

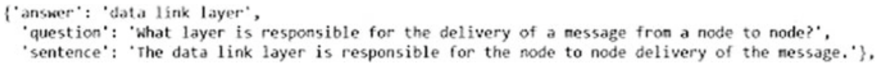
4 Results and Evaluation

As seen in Fig. 5, a Wh-question is generated by our model with the ans as “data link layer”. The quality of question is really well considering the original sentence as “the data link layer is responsible for the node to node delivery of the message” and the corresponding question as “what layer is responsible for the node to node delivery of the message?”.

Figure 6 represents a sample fill in the blanks type of question with the original sentence being “the data in the transport layer is referred to as segments” and corresponding blank in the question is at the position of “data”. Even for the MCQ question, shown in Fig. 7, the answer for the question “transport layer provides services to and takes services from network layer” is “application layer” with confusing distractors like “http”. Hence, we can see that our pipeline is giving satisfactory results for all types of question generation (Fig. 8).

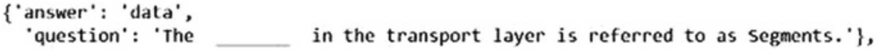
We propose our own method for evaluating our system involving human reference. Human-System Comparison—In this, we took a set of questions generated by our system and another set generated by a human, and we gave these two sets to a third person who was asked to identify the questions generated by the system and those by the human. When he was not able to correctly identify the machine generated questions, we concluded that our system is at par with humans.

We performed this method by taking three corpuses from the fields of software engineering, computer networks and object oriented programming. Then, we used our pipeline to generate three questions each of fill in the blanks, multiple choice



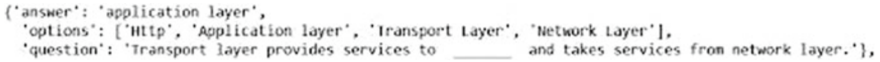
```
{'answer': 'data link layer',
  'question': 'What layer is responsible for the delivery of a message from a node to node?',
  'sentence': 'The data link layer is responsible for the node to node delivery of the message.'},
```

Fig. 5 Sample output for Wh-based question, generated from the given sentence and answer phrase



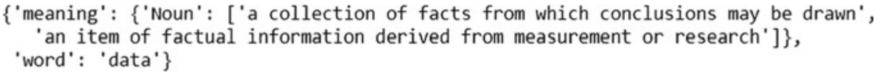
```
{'answer': 'data',
  'question': 'The ________ in the transport layer is referred to as Segments.'},
```

Fig. 6 Sample output for Fib type questions



```
{'answer': 'application layer',
  'options': ['Http', 'Application layer', 'Transport Layer', 'Network Layer'],
  'question': 'Transport layer provides services to ______ and takes services from network layer.'},
```

Fig. 7 Sample output for MCQ type questions



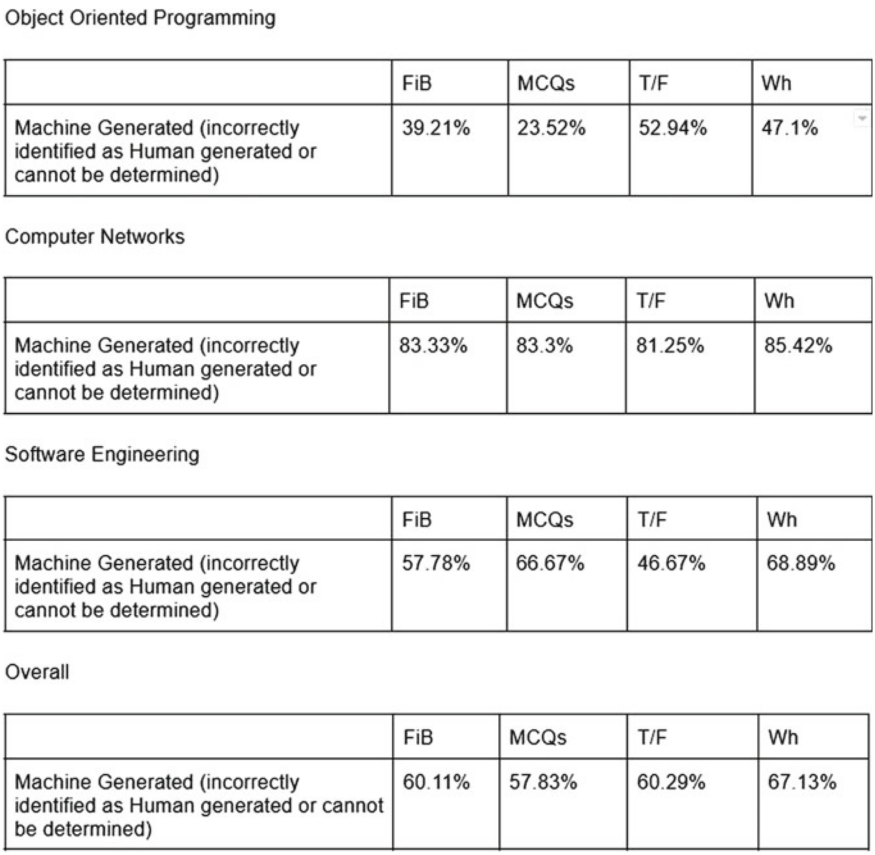
```
{'meaning': {'Noun': ['a collection of facts from which conclusions may be drawn',
    'an item of factual information derived from measurement or research']},
  'word': 'data'}
```

Fig. 8 Sample output for flashcard of the word “data”

questions, Wh-based questions and two questions of true or false for each of the corpuses. Then, we mixed these questions with an equal number of human generated questions for each type and each corpus.

Later, we created a survey wherein users were asked to identify whether the question is human generated or machine generated or if it could not be determined. We got responses from around 45 people in this survey. The results of this process are shown in Fig. 9.

From Fig. 9, we could see that for the corpuses of computer networks and software engineering, the questions generated from our pipeline were very difficult to be identified as generated by a machine. Around 83% responses from the subject of computer networks and 60% responses from the subject of software engineering shows that the questions generated from machines were very much similar to those generated by a human. For creating the overall table, we took the average of the percentage of incorrect choices for all the question types for all three corpuses.



Object Oriented Programming

	FiB	MCQs	T/F	Wh
Machine Generated (incorrectly identified as Human generated or cannot be determined)	39.21%	23.52%	52.94%	47.1%

Computer Networks

	FiB	MCQs	T/F	Wh
Machine Generated (incorrectly identified as Human generated or cannot be determined)	83.33%	83.3%	81.25%	85.42%

Software Engineering

	FiB	MCQs	T/F	Wh
Machine Generated (incorrectly identified as Human generated or cannot be determined)	57.78%	66.67%	46.67%	68.89%

Overall

	FiB	MCQs	T/F	Wh
Machine Generated (incorrectly identified as Human generated or cannot be determined)	60.11%	57.83%	60.29%	67.13%

Fig. 9 Survey results for machine generated questions

5 Conclusion

A lot of methodologies and different kinds of approaches have been proposed in the field of natural language processing that impact the amount of computation and speed of inference. We have performed extensive literature survey to understand and compare existing solutions and identify gaps in them. We came up with our custom NLP pipeline that involves text summarization, keyword extraction, sentence selection and distractor generation using the latest state-of-the-art models which proved to be a major improvement over existing methods. We also came up with our own metrics for evaluating the accuracy of our pipeline involving human intervention. With our own custom pipeline, we were able to generate good quality questions, and with our metric, we found that around 60–80% of our survey responses identified machine generated questions as human generated or could not be determined. Our research may prove to be a stepping stone for further research in this field. Our research can be integrated in systems that might eventually help in testing student's attentiveness during lectures.

References

1. Sotola L, Marcus C (2021) Regarding class quizzes: a meta-analytic synthesis of studies on the relationship between frequent low-stakes testing and class performance. Educ Psychol Rev 33. https://doi.org/10.1007/s10648-020-09563-9
2. Singh Bhatia A, et al (2013) Automatic generation of multiple choice questions using Wikipedia. In: Maji P et al (eds) Pattern recognition and machine intelligence, Springer, pp 733–738. https://doi.org/10.1007/978-3-642-45062-4
3. Pabitha P, Mohana M, Suganthi S, Sivanandhini B (2014) Automatic question generation system. Int Conf Recent Trends Inform Technol 2014:1–5. https://doi.org/10.1109/ICRTIT.2014.6996216
4. Aithal SG, Rao AB, Singh S (2021) Automatic question-answer pairs generation and question similarity mechanism in question answering system. Appl Intell
5. Srivastava A, et al (2020) Questionator-automated question generation using deep learning. In: 2020 international conference on emerging trends in in-formation technology and engineering (Ic-ETITE), 2020, pp 1–5. IEEE Xplore. https://doi.org/10.1109/ic-ETITE47903.2020.212
6. Liu Z, et al (2020) Semantics-reinforced networks for question generation. ECAI. Semantic Scholar. https://doi.org/10.3233/FAIA200330.7
7. Rohde T, et al (2021) Hierarchical learning for generation with long source sequences. http://arxiv.org/abs/2104.07545
8. Wang Z, et al (2018) QG-Net: a data-driven question generation model for ed-ucational content. In: Proceedings of the fifth annual ACM conference on learning at scale, ACM, pp 1–10. https://doi.org/10.1145/3231644.3231654
9. Savelieva A, et al (2020) Abstractive summarization of spoken and written instructions with BERT. http://arxiv.org/abs/2008.09676
10. Metatext, https://metatext.io/models/sentence-transformers-distilbert-base-nli-stsb-mean-tokens
11. Reimers N, Iryna G (2019) Sentence-BERT: sentence embeddings using siamese BERT-networks. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, pp 3982–3992

12. Rael C, Shazeer NM, Roberts A, Lee K, Narang S, Matena M, Zhou Y, Li W, Liu PJ (2020) Exploring the limits of transfer learning with a Unified text-to-text transformer. http://arxiv.org/abs/1910.10683
13. Lelkes AD, et al (2021) Quiz-style question generation for news stories. http://arxiv.org/abs/2102.09094
14. Sahrawat D, et al (2020) Keyphrase extraction as sequence labeling using contextualized embeddings. In: Jose J et al (eds) Advances in information retrieval. ECIR 2020. LectureNotes in Computer Science, vol 12036. Springer, Cham. https://doi.org/10.1007/978-3-030-45442-541
15. Nikzad-Khasmakhi N, Feizi-Derakhshi M, Asgari-Chenaghlu M, Balafar M, Feizi-Derakhshi A, Rahkar-Farshi T, Ramezani M, Jahanbakhsh-Nagadeh Z, Zafarani-Moattar E, Ranjbar-Khadivi M (2021) Phraseformer: multi-modal key-phrase ex-traction using transformer and graph embedding. http://arxiv.org/abs/2106.04939. Das B, et al (2021) Automatic question generation and answer assessment: a survey. Res Prac Technol Enhan Learn 16(1):5. BioMed Central. https://doi.org/10.1186/s41039-021-00151-1
16. Melekhov I, et al (2016) Siamese network features for image matching. In: 2016 23rd international conference on pattern recognition (ICPR), IEEE, 2016, pp 378–383. https://doi.org/10.1109/ICPR.2016.7899663
17. Vaswani A, et al (2017) Attention is all you need. Google Res. https://arxiv.org/pdf/1706.03762.pdf
18. Devlin J, et al (2019) BERT: pre-training of deep bidirectional transformers for language understanding. http://arxiv.org/abs/1810.04805
19. Killawala A, et al (2018) Computational intelligence framework for automatic quiz question generation. In: 2018 IEEE international conference on fuzzy systems (FUZZ-IEEE), IEEE, pp 1–8. https://doi.org/10.1109/FUZZ-IEEE.2018.8491624
20. Sreelakshmi AS, et al. (2019) A question answering and quiz generation Chatbot for education. In: 2019 Grace Hopper Celebration India (GHCI), 2019, pp 1–6. IEEE Xplore. https://doi.org/10.1109/GHCI47972.2019.9071832
21. Kurdi G, et al (2020) A systematic review of automatic question generation for educational purposes. Int J Arti Intell Educ 30(1):121–204. https://doi.org/10.1007/s40593-019-00186-y
22. Susanti Y, et al (2018) Automatic distractor generation for multiple-choice english vocabulary questions. Res Prac Technol Enhan Learn 13(1):15. BioMed Central. https://doi.org/10.1186/s41039-018-0082-z
23. Automatic Question Generation From Passages. Springerprofessional.De, https://www.springerprofessional.de/en/automatic-question-generation-from-passages/16186598. Accessed 9 Jul 2021
24. Nwafor CA, Onyenwe IE (2021) An automated multiple-choice question generation using natural language processing techniques. Int J Nat Lang Comp 10(2):1–10. https://doi.org/10.5121/ijnlc.2021.10201
25. Kriangchaivech K, Wangperawong A (2019) Question gener-ation by transformers. http://arxiv.org/abs/1909.05017
26. Singh J (2018) Encoder-decoder architectures for generating questions. Proc Comp Sci 132:1041–1048. www.sciencedirect.com, https://doi.org/10.1016/j.procs.2018.05.019
27. Qi W, et al (2020) ProphetNet: predicting future N-gram for sequence-to-sequence pre-training. http://arxiv.org/abs/2001.04063
28. Aithal SG, et al (2021) Automatic question-answer pairs generation and ques-tion similarity mechanism in question answering system. Appl Intell. https://doi.org/10.1007/s10489-021-02348-9

Multiclass Image Classification of COVID-19 Chest X-ray Scans Using Deep Learning

Laya Rathod, Harsh Jain, Jayakumar Kaliappan, and C. Suganthan

Abstract The novel coronavirus is a family of animal transferred viruses that can cause illness in humans. This virus took over the world in 2019 and WHO deemed it as an epidemic naming it as COVID-19. A lot of research has gone in the prediction of the trends and classification of cases using various machine learning and deep learning techniques. With the outbreak of this pandemic, efficient detection of the disease at a faster rate has become very crucial. This study proposes a convolutional neural network (CNN)-based deep learning approach for classification of COVID-19 positive cases from normal cases using X-Ray radiology scans of the patients. The model consists of a large custom dataset of images extracted from an open source dataset and is then trained using our proposed model. Different optimizer algorithms are also compared in order to check which one of them gives the most accuracy. The model is further tested using the categorical accuracy metrics and then a graphical analysis of the results is provided. A comparative study is also conducted with an already existing support vector machines (SVM) model. The images were trained according to three classifications: normal, COVID infected, and viral pneumonia infected patients. The main objective of our research is to help further the research in early diagnosis of COVID-19 using modern deep learning techniques.

Keywords Convolutional neural network · COVID-19 · Deep learning · Image classification · Support vector machines

1 Introduction

The novel coronavirus or the COVID-19 was first identified in Wuhan, China. Further, it didn't take long for it to spread to the other parts of the country, and eventually, the whole world. COVID-19 is an infectious disease which can easily be transmitted through coming in direct contact (coughing or sneezing) with an infected person. Even though the symptoms for COVID-19 are very similar to the common cold, the

L. Rathod · H. Jain · J. Kaliappan (✉) · C. Suganthan
School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, India
e-mail: jayakumar.k@vit.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_38

virus is capable of causing other chronic respiratory illnesses as well. There have been a total of 167,531,200 cases that the world has experienced till May, 24, 2021, of which, 15,505,637 were still active.

Currently, a special test called the reverse transcription polymerase chain reaction (RT-PCR) is conducted to test the presence of the virus, but it generally takes 24 h to get the result. Some studies show that when a person is infected with the virus, there are some significant changes that could be noticed in the infected person's chest. If these changes can be identified well in time, it might lead to huge clinical benefits. Since, getting X-rays is not only cheap but also is a less time consuming process, this method could aid doctors in identifying and classifying any abnormalities that may be present. It could further be implemented as a screening stage for the RT-PCR test, thus saving time and resources as shown in Fig. 1.

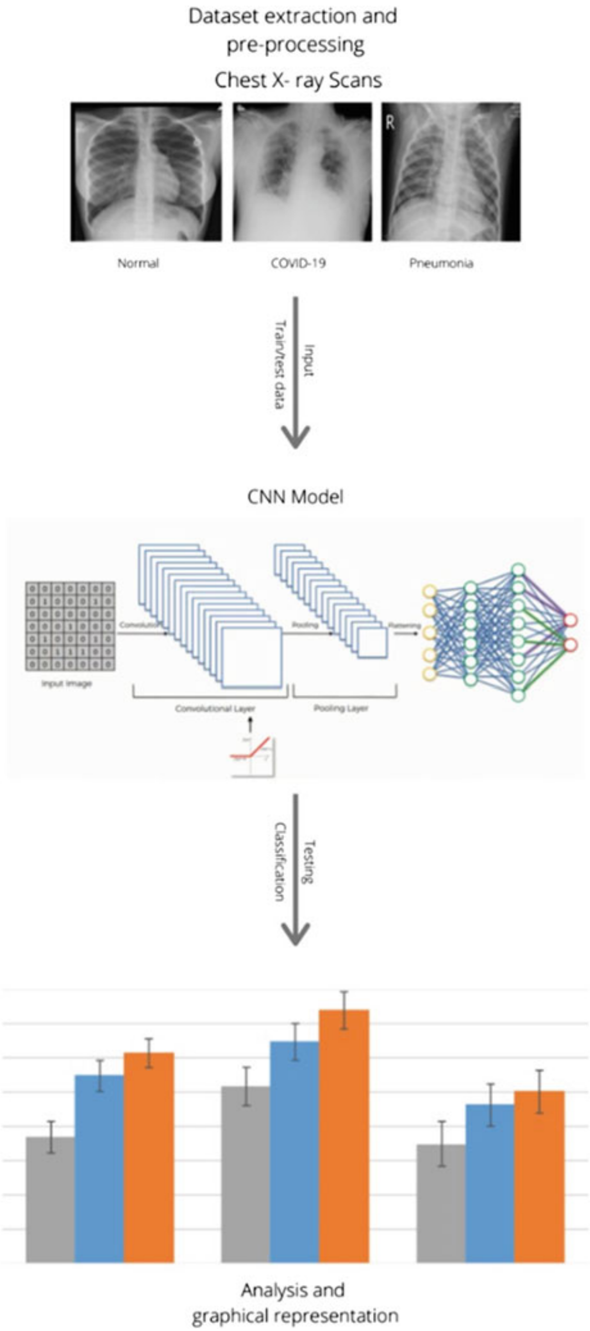
The aim of our study is to create an efficient and best possible system that will help distinguish these changes and predict if a person has been infected by COVID-19 or not. A 23-layer CNN, consisting of six different kinds of layers, is proposed. The main objective is to figure out the most efficient optimizer that would give the best accuracy, to check if increasing the number of classes and the scale of the datasets will make the efficiency and accuracy of the model vary and whether we can use the same model to classify multiple classes.

2 Literature Survey

It has been noticed through medical research that chest X-ray is an imaging technique that can play a vital role in the detection of COVID-19. More and more large-scale datasets are being made available for research in these field and convolutional neural networks have emerged to be great tools for image classification. The research field is new and the time period is less, hence the biggest limitation is the availability of labelled medical images, which make it easier to model for classification [1].

Currently, even though the reverse transcription-polymerase chain reaction (RT-PCR) shows a very low sensitivity and is relatively a time consuming process, it is still the most widely used technique for the detection of COVID-19 [2]. It has been demonstrated how machine learning techniques, which have been used so often for detection of potential risk factors, can also be implemented to predict the number of cases of the ongoing COVID-19 threat. ML modelling was used to analyze the exponential smoothing risk factors. [3] A system which uses blood testing to support the diagnosis of COVID-19 was also implemented where the parameters are obtained from hemogram and the biochemical tests [4]. The healthcare sector is overburdened and the need for fast and efficient detection procedures is the need of the hour. Through recent medical research it has been established that detection of COVID-19 by means of X-ray and CT scans is a possibility [5]. The countries with shortage of proper equipment and testing kits have also been seen relying on CT scan images and x-rays. Research also shows that in order to make the COVID detection process

Fig. 1 Graphical representation of proposed experiment



efficient and faster, a combination of RT-PCR along with image featuring models can be used [6].

Medically speaking, there have been a lot of research to find out the key features to look for when detecting COVID chest X-ray scans from a normal scan. The COVID X-rays were found to have an abnormal increase in opacification near the lung region. This was coupled with reticulation interstitial thickening due to a collection of small opacities in a linear pattern. Ground glass is a radiological term that is used to define a hazy opaque lung appearance. These opacities may become dense in some cases that might obscure bronchial walls and vessels. We use feature extraction and modern image processing techniques to enhance this opacity features and train the model with a large number of images for detection of COVID-19 scans in comparison to Normal scans.

The recent large scale data mining and deep learning techniques such as CNNs are applied for rapid detection of the disease. This research saw the progress in development of a CNN model called CoroDet. It serves for accurate detection for multi-class classification. The 22-layer architecture of the CNN model that was proposed could classify with a max accuracy of 99.1% for the two classes [7].

The shortcoming of most researches in this field was that the dataset was used was small and only as far as the knowledge goes. Future work includes working on larger and more comprehensive data as we move forward into more research in this field. This can be a big step in providing a supportive alternative to tackle the problem of testing kit scarcity for detection of COVID-19 [8].

Research has also shown that it is also possible to differentiate whether the problems caused is because of COVID infection or any other pulmonary disease. Supervised machine learning techniques can be used for the same and they consists of mainly two phases: training and testing. The training phase discriminates between the X-ray images of COVID-19 images and other pathologies, whereas the testing phase then evaluates the performance of the model built in the training phase [9]. Data pre-processing is a major aspect here since data-imbalances can sway the predictions. Hyperparameter optimization using grid search algorithm provided more accurate predictions as compared to traditional random forest model. These models can integrated into real mobile healthcare applications. Rapid detection and screening for COVID-19 infections can be made possible through such semi-autonomous diagnostic systems [10].

Studies also showed that multiple feature extraction methods using support vector machines (SVM) can be used in order to classify the extracted features of different classes [11]. The SVM model can also be used to produce real-time forecasting. Data attributes which included confirmed, deceased, recovered COVID-19, and location wise data were collected. And then to explore the impact on recovery, deceased and identification, SVM was used. [12] A major limitation for such models is the availability of labeled medical images. A solution to this is the concept of transfer learning. It is an effective mechanism and shows promise in future results. The main solution is to transfer knowledge from conventional object detection tasks to specific domains. The research explores a kind of deep CNN called DeTraC which stands for decompose, transfer, and compose. An advantage of using such a model is that it can easily deal with irregularities in the image datasets. It uses a class decomposition mechanism to analyze class boundaries [13].

Further, in order to enable the identification of false prediction or unknown cases, Monte-Carlo Drop weights Bayesian convolutional neural network was used on publically available chest X-ray dataset. A pre-trained ResNet50V2 model was used for transfer learning [14].

Even though COVID-19 detection techniques based on CNN and various other models can become well established, they are still in the testing phases. There is a great room for improvement in this domain. These deep learning techniques are becoming more and more popular and can be easily associated with well-established medical techniques [15].

3 Proposed Methodology

Figure 2 explains the various steps that we underwent for the experiment. The first step was to extract the dataset from open-source platforms. We found clear images which were classified into COVID, Normal and pneumonia chest X-ray scans. Only PA (Posterior Anterior) view scans were extracted since they are most relevant for our feature extraction. The second step consisted of data pre-processing and labeling. The dataset was first given proper labels and was sorted into training and testing datasets of 80-20% and further into directories of the three categories. Next, the generator functions of Keras were used to take the inputs in real-time batches where we adjusted the size, zoom range, and other parameters in order to get the most optimal training dataset. The third step is the most important where we tested a number of permutations and combinations of different layers and optimizers in order to find the most accurate CNN model for classification. The fourth step included the training phase using our final proposed model. We used categorical parameters to get results on the training data. Lastly, the learning curves for model accuracy and loss were plotted along with other parameters. The proposed model was compared with previously tested SVM models in order to find the relevance of our model.

3.1 Dataset

Various open-source datasets were scanned for X-Ray images of the three categories: COVID-19 infected, viral pneumonia infected and normal cases.

A large open-source COVID-19 radiography database on Kaggle was identified which consisted of 3616 COVID-19 positive X-rays, 1345 viral pneumonia x-rays, and 10,192 normal x-ray scans. All the images in the dataset were of the posterior anterior (PA) view, the best suited for our model [16–18].

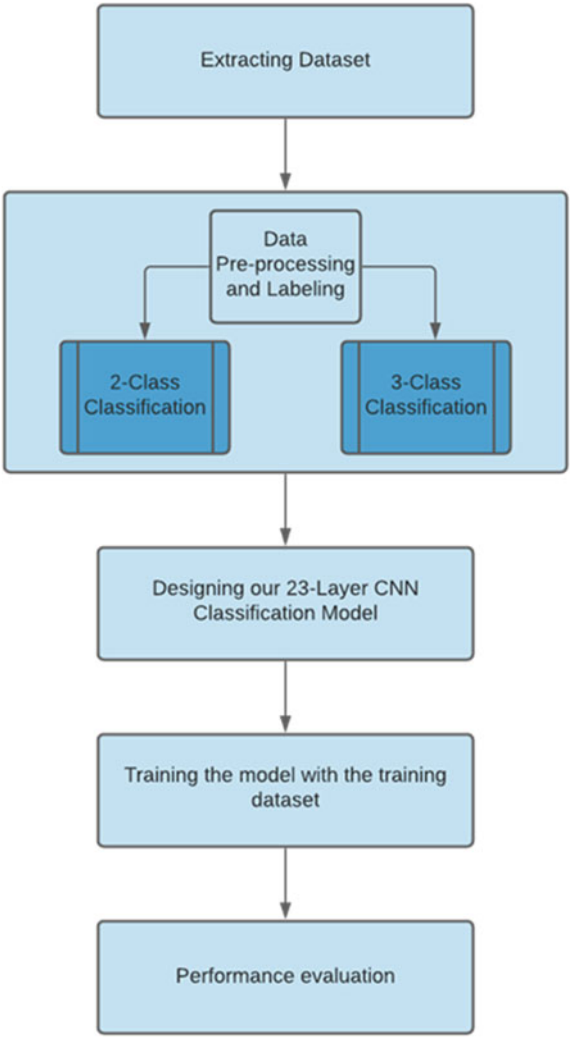

Extracting Dataset
Data Pre-processing and Labeling
2-Class Classification
3-Class Classification
Designing our 23-Layer CNN Classification Model
Training the model with the training dataset
Performance evaluation

Fig. 2 A flowchart of our proposed methodology

3.2 Data Labeling and Structuring

Since the original dataset was unevenly distributed, we extracted our own customized dataset from the same. These images were then stored in categorical folders of testing (20%) and training data (80%). The directory structure for the 3-class dataset included training (total: 2400 images) and testing (total: 600 images) directories which were then further divided into three categorical directories of COVID, normal and pneumonia, each containing equal number of images. The 2-class dataset consisted of a similar structure but only contained the categorical directories of COVID and normal (Table 1).

Table 1 Dataset structure

		3-Class	2-Class
Training	COVID	800	800
	Normal	800	800
	Pneumonia	800	–
Testing	COVID	200	200
	Normal	200	200
	Pneumonia	200	–

3.3 Data Pre-processing

We used the Image Data Generator function to augment the images in real time. This helps in memory utilization since the process runs while the model is training and no excess memory is required to store the processed dataset. It also performs fewer transformations before going through the model.

Further, the images were rescaled by dividing them by a factor of 255 so that their spatial values ranged between 0 and 1. The zoom and shear range was set to 0.2 and a horizontal flip augmentation of the images was performed for the training dataset.

3.4 CNN Model

We used Keras and Tensorflow framework for creating our proposed CNN model. This consisted of a 23-layer architecture which comprised of six basic components, namely, convolutional layer, batch normalization layer, pooling layer, dropout layer, flatten layer, and dense layer.

The purpose of the convolutional layer is to help the model learn. With the help of certain parameters, filters and kernels, this layer performs most of the computations in terms of the convolutional process, which is the most important for a CNN model. In this layer, image segmentation procedures are performed to distinguish differences and similarities between the images. Our model uses this layer to extract high level features like edges of the rib cage and distinguishes it from the lung region, where the infection could be detected. We used eight conv2D layers in our model where the kernel size for the first four layers was taken as three and the last four as five. The number of output filters exponentially increased by a factor of 2 from 16 till 128. In the model, the conv2D layers were used in pairs. The input shape taken was (244, 244, 3) corresponding to the size of the images in our dataset. The activation function used was ‘relu’.

We used a batch normalization layer after each pair of the convolutional layer to maintain the mean output near 0 and the standard deviation near 1. The role of this layer is to help every other layer of the model to learn more independently.

The purpose of the pooling layer is to prevent overfitting by helping in reduction of the image size when the volume is too large. We used the maxpool2D layer with the stride set to (2, 2). This takes the largest value from the feature map received from the previous layer. The stride allows us to specify the movement of the filter over the image.

The dropout layer also helps us to prevent overfitting of the data. We set the rate to 0.25 which means that this fraction of input units were dropped after this layer for the maximum optimization.

Before passing the data to the next layer, the flatten layer helps us in reducing the dimensionality of the pool feature map into a single column.

We used three consecutive dense layers with the activation function 'softmax'. These are fully connected layers that flatten the input from the previous matrix into a vector. This is the layer which is able to decide which features best match a particular class. This is where the classification takes place. For the 2-class structure, the last layer will have two units and similarly the 3-class structure will have three.

Figure 3 shows the model summary of our CNN architecture. The total parameters in the model are 7,735,891 out of which 7,735,411 are trainable and 480 are non-trainable.

4 Results and Discussion

4.1 Training the Model

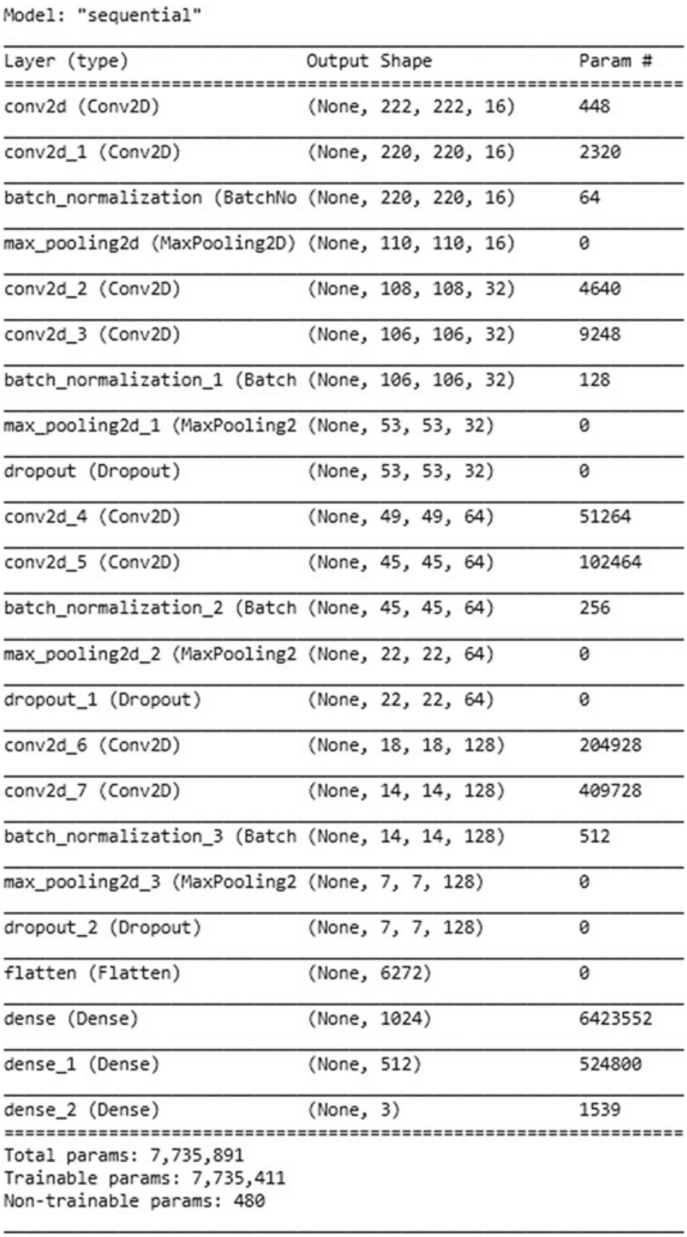
We performed two sets of experiments, one for the 3-class and one for the 2-class image classification. The same model is used to classify both the datasets. The three classes were taken as COVID, normal, and Pneumonia.

The training was carried out on Google Colab for its GPU capabilities and the model was implemented using the Tensorflow and Keras frameworks.

We first chose three Keras optimizers for our experiment, namely, RMS, Adam, and Adamax. We ran simulations for all the three individually, under the same conditions and compared the results to choose which optimizer is best suited for our model. The experiment was run for 100 epochs with a batch size set to 16. The steps per epoch and validation steps were tested for multiple values, but it was noticed that optimal results were obtained when both were kept at 16 each.

The RMS optimizer was best suited for our model. The RMS prop algorithm was used to define the optimizer where the learning rate was kept at 0.0001 and the rho at 0.95. The discounted average of the gradient squares is maintained through this algorithm.

The loss function set was 'catergorical_crossentropy' and the metrics as 'categorical_accuracy'. The graphs of accuracy and loss for training and testing were plotted to observe the learning curve.



```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
conv2d (Conv2D)              (None, 222, 222, 16)      448       
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 220, 220, 16)      2320      
_________________________________________________________________
batch_normalization (BatchNo (None, 220, 220, 16)      64        
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 110, 110, 16)      0         
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 108, 108, 32)      4640      
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 106, 106, 32)      9248      
_________________________________________________________________
batch_normalization_1 (Batch (None, 106, 106, 32)      128       
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 53, 53, 32)        0         
_________________________________________________________________
dropout (Dropout)            (None, 53, 53, 32)        0         
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 49, 49, 64)        51264     
_________________________________________________________________
conv2d_5 (Conv2D)            (None, 45, 45, 64)        102464    
_________________________________________________________________
batch_normalization_2 (Batch (None, 45, 45, 64)        256       
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 22, 22, 64)        0         
_________________________________________________________________
dropout_1 (Dropout)          (None, 22, 22, 64)        0         
_________________________________________________________________
conv2d_6 (Conv2D)            (None, 18, 18, 128)       204928    
_________________________________________________________________
conv2d_7 (Conv2D)            (None, 14, 14, 128)       409728    
_________________________________________________________________
batch_normalization_3 (Batch (None, 14, 14, 128)       512       
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 7, 7, 128)         0         
_________________________________________________________________
dropout_2 (Dropout)          (None, 7, 7, 128)         0         
_________________________________________________________________
flatten (Flatten)            (None, 6272)              0         
_________________________________________________________________
dense (Dense)                (None, 1024)              6423552   
_________________________________________________________________
dense_1 (Dense)              (None, 512)               524800    
_________________________________________________________________
dense_2 (Dense)              (None, 3)                 1539      
=================================================================
Total params: 7,735,891
Trainable params: 7,735,411
Non-trainable params: 480
_________________________________________________________________
```

Fig. 3 Model summary

5 Results and Discussion

In this section, we have mentioned the training and validation accuracy metrics in Tables 2 and 3 for the 3-class and 2-class models, respectively. We observed that in both the models, the RMS and the Adamax optimizers performed better than Adam optimizer, RMS giving the best results of 0.9082 and 0.9875 validation accuracy in the 3-class and 2-class models, respectively.

It was also observed that increasing the number of classes led to a decrease in the accuracy of the models.

Figure 4 shows the graph plotted for the training and testing accuracy against the number of epochs for the RMS optimizer. We observed that the training accuracy remained fairly constant throughout, whereas the testing accuracy increased and then stabilized around the range of 40 epochs.

Figure 5 shows the graph plotted for the training and testing loss against the number of epochs for the RMS optimizer. We observed that the training loss, again, remained fairly constant throughout, whereas the testing loss decreased and then stabilized around the range of 40 epochs.

In Table 4, we have shown the results of the comparison of all the optimizers used for the 2-class model with the already existing SVM model. We observed that the SVM model gave a 2-class accuracy of 0.96 whereas our proposed 2-class RMS model gave the best accuracy of 0.9875.

Table 2 Categorical accuracy metric results for 3-class model

Metrics	Optimizers		
	rms	adam	adamax
Training_loss	0.2687	0.7306	0.5148
Training_accuracy	0.8983	0.7500	0.8450
Validation_loss	0.3216	0.5942	0.3982
Validation_accuracy	0.9082	0.7721	0.8885

Table 3 Categorical accuracy metric results for 2-class model

Metrics	Optimizers		
	rms	adam	adamax
Training_loss	0.1151	0.2499	0.1765
Training_accuracy	0.9645	0.914	0.9465
Validation_loss	0.0528	0.1793	0.0458
Validation_accuracy	0.9875	0.9275	0.98

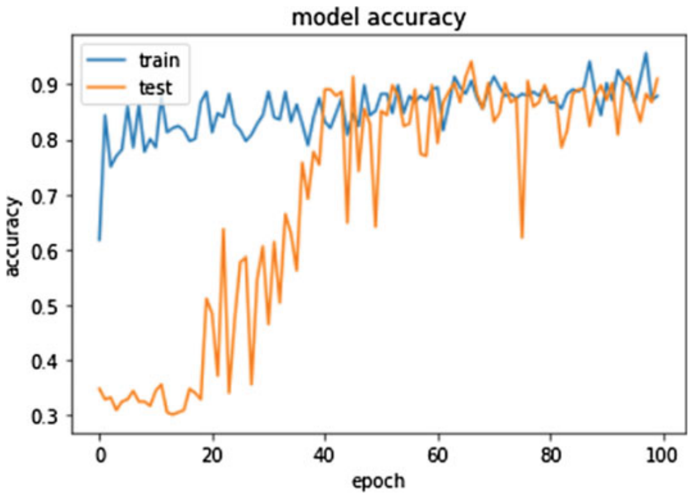


Fig. 4 Class model accuracy for 100 epochs for the rms optimizer

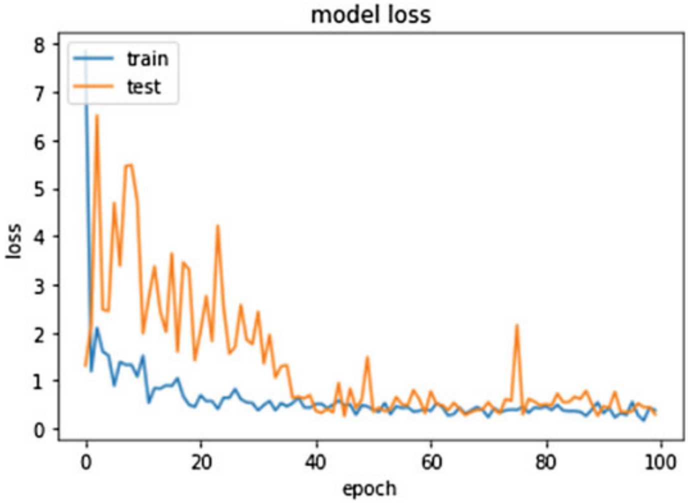


Fig. 5 Class model loss for 100 epochs

Table 4 Categorical accuracy metric results for 2-class model

	Model			
	SVM	rms	adam	adamax
Accuracy	0.9600	0.9875	0.9275	0.9800

6 Conclusion

Our study focusses on research of deep learning application in the field of medical science related to COVID-19 and helps by providing a technique which can be used, if not as an alternative, but as an additional method for identifying and classifying any abnormalities that might be present while diagnosing any pulmonary diseases. We proposed a 23-layer CNN architecture which was tested for 3-class (COVID, normal, and pneumonia) and 2-class (COVID, Normal) models. Multiple optimizers (RMS, Adam, Adamax) were used and the RMS optimizer provided us with the best accuracy. This model was further compared with the already existing machine learning based SVM model and, even here, our proposed RMS model showcased better accuracy. It was noticed that when the model was tested on larger datasets, the 2-class accuracy was approximately the same, whereas the 3-class accuracy decreased to 85–90%. Further research can be done to optimize the model in order for it to give constant performance results with larger datasets for both 2-class and 3-class classification.

Through this study, we were able to successfully implement a deep learning-based CNN model, fully capable of multi-class image classification of chest X-ray scans. Future research can revolve around the implementation of transfer learning in the same field. Further study could also be done on how we can maintain the model accuracy even while scaling the dataset.

References

1. Aljondi R, Alghamdi S (2020) Diagnostic value of imaging modalities for COVID-19: scoping review. J Med Internet Res 22(8):e19673
2. Xie X, Zhong Z, Zhao W, Zheng C, Wang F, Liu J (2020) Chest CT for typical coronavirus disease 2019 (COVID-19) pneumonia: relationship to negative RT-PCR testing. Radiology 296(2):E41–E45
3. Mojjada RK, Yadav A, Prabhu AV, Natarajan Y (2020) Machine learning models for covid-19 future forecasting. Mater Today Proc
4. de Freitas Barbosa VA, Gomes JC, de Santana MA, de A. Jeniffer E, de Souza RG, de Souza RE, dos Santos WP (2021) Heg. IA: an intelligent system to support diagnosis of Covid-19 based on blood tests. Res Biomed Eng 1–18
5. Lee EYP, Ng MY, Khong PL (2020) COVID-19 pneumonia: what has CT taught us? Lancet Infect Dis 20(4):384–385
6. Kong W, Agarwal PP (2020) Chest imaging appearance of COVID-19 infection. Radiol Cardiothoracic Imaging 2(1):e200028
7. Hussain E, Hasan M, Rahman MA, Lee I, Tamanna T, Parvez MZ (2021) CoroDet: a deep learning based classification for COVID-19 detection using chest X-ray images. Chaos, Solitons Fractals 142:110495
8. Maguolo G, Nanni L (2021) A critic evaluation of methods for covid-19 automatic detection from x-ray images. Inform Fusion
9. Brunese L, Martinelli F, Mercaldo F, Santone A (2020) Machine learning for coronavirus covid-19 detection from chest x-rays. Proc Comp Sci 176:2212–2221

10. Kaliappan J, Srinivasan K, Mian Qaisar S, Sundararajan K, Chang CY (2021) Performance Evaluation of Regression Models for the Prediction of the COVID-19 Reproduction Rate. Front. Public Health, 9
11. Barstugan M, Ozkaya U, Ozturk S (2020) Coronavirus (covid-19) classification using CT images by machine learning methods. arXiv preprint arXiv:2003.09424
12. Singh V, Poonia RC, Kumar S, Dass P, Agarwal P, Bhatnagar V, Raja L (2020) Prediction of COVID-19 corona virus pandemic based on time series data using support vector machine. J Discrete Math Sci Cryptogr 1–15
13. Abbas A, Abdelsamea MM, Gaber MM (2021) Classification of COVID-19 in chest X-ray images using DeTraC deep convolutional neural network. Appl Intell 51(2):854–864
14. Jayakumar K, Revathi T, Karpagam S (2015) Fusion of Heterogeneous Intrusion Detection Systems for Network Attack Detection. Sci World J
15. Apostolopoulos ID, Mpesiana TA (2020) Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks. Phys Eng Sci Med 43(2):635–640
16. Rahman T, Khandakar A, Qiblawey Y, Tahir A, Kiranyaz S, Kashem SBA, Islam MT, Maadeed SA, Zughaier SM, Khan MS, Chowdhury ME (2020) Exploring the effect of image enhancement techniques on COVID-19 detection using chest X-ray images.
17. Chowdhury MEH, Rahman T, Khandakar A, Mazhar R, Kadir MA, Mahbub ZB, Islam KR, Khan MS, Iqbal A, Al-Emadi N, Reaz MBI, Islam MT (2020) Can AI help in screening viral and COVID-19 pneumonia? IEEE Access 8:132665–132676
18. Tawsifur R, Chowdhury M, Khandakar A (2021) https://www.kaggle.com/tawsifurrahman/covid19-radiography-database (version 4) [online]

Plant Disease Detection Using Image Processing and Machine Learning

Venus Patel, Noopur Srivastava, and Manish Khare

Abstract In agriculture, plant disease and its precise detection is an important task. Researchers have attempted several methods to automate disease detection using the latest tools and techniques of image processing and machine learning. In this paper, a semi-automatic system is designed to detect two diseases of soybean (glycine max) named mosaic virus and leaf spot. We used the k-means clustering approach to extract the combined colour and texture features from the diseased area of soybean leaves and classified it using the KNN algorithm. The observations are compared with the existing work to validate the efficacy of the proposed approach. Moreover, visual inspections of the leaf sample also prove the suitability of the proposed system for detection and classification.

Keywords Plant disease detection · Image processing · Machine learning · KNN algorithm

1 Introduction

India is considered an agricultural nation because around 70% of the total inhabitants' income relies on farming. However, the plant life and the yields are pretentious via the ailment which directs the abridged amount of eminence and some agricultural goods. The learning of plant infections describes the optically visible prototypes. It checks the fitness and diseases of vegetation. The specialists monitor the plant infection with their bare eyes and recognize the kind of infection the crop is suffering

V. Patel
Ganpat University, Mehsana, India

N. Srivastava
Shri Ramswaroop Memorial University, Lucknow-Deva Road, Barabanki, India

M. Khare (✉)
Dhirubhai Ambani Institute of Information and Communication Technology, (DA-IICT), Gandhinagar, India
e-mail: mkharejk@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_39

from. For performing this scrutiny, a group with many specialists are prepared and incessantly observe the crop. This process is relatively costly when the fields are enormous. At times, the farmers do not find themselves familiar with superior technology, which becomes the reason for the setback of infection discovery. In these cases, getting in touch with specialists is not a simple job; it is also costly and time consuming. Alternatively, if superior techniques are utilized, the discovery will turn out to be inexpensive and less time consuming. Usually, several ordinary diseased plants suffer from brown and yellow spots, untimely or belatedly singe, and many other fungal infections. In [1], Bashir and Sharma provided a picture processing method for detecting plant infections. Picture processing is the method utilized to compute and evaluate the infected area and concludes the alteration in the colour of that diseased area [2].

1.1 Essential Steps for Disease Detection

The fundamental phases involved in the detection of plant infections are specified in Fig. 1. The primary stage involves picture acquisition or attainment, in which the pictures are taken from the atmosphere with the help of the digital camera. The secondary stage is the picture pre-processing, in which the descriptions are retrieved from the obtained pictures for additional scrutiny. After this, many investigative methods are used to categorize pictures according to the same infections [2]. The basic procedure of the proposed vision-based disease detection algorithm is shown in Fig. 1, and its phases are explained as follows:

(a) Image Acquisition: The plant leaf picture is obtained through a camera, and the views are in red, green, and blue. It generates a colour change of the RGB leaf picture, and hereafter, equipment-independent gap alteration is executed.

(b) Image Pre-processing: For removing unwanted noises from the pictures, different pre-processing methods are initiated. Picture clipping which engages cropping of the leaves pictures contains a required picture area. Levelling filter is

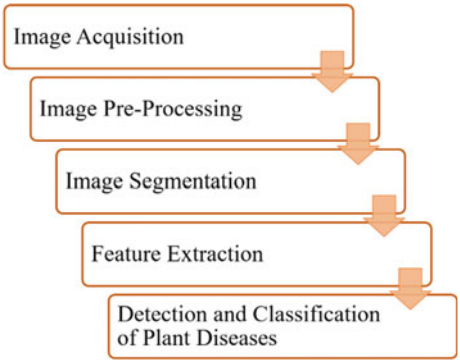


Fig. 1 Basic procedure of the vision-based disease detection algorithm

utilized for picture levelling. Picture improvement is accountable for the growth of colour contrast.

(c) Image Segmentation: Segmentation or sectioning denotes the division of the pictures into different sections encompassing similar descriptions and qualities. It can be implemented with the help of other means such as the Otsu process, k-means clustering, and alteration of RGB pictures.

(d) Feature Extraction: It plays an essential character in entity discovery. It is utilized in numerous picture processing functions, including colour, consistency, morphology, boundaries, etc. These are the descriptions of plant infection recognition. The morphological characteristics are superior in comparison with other explanations. At this time, consistency indicates the colour of the pictures, unevenness, and rigidity of the photographs. It is also utilized for the recognition of diseased plants and their yields.

(e) Classification and Detection: The final phase of plant infection recognition is the classification of the plants following the infections. For this purpose, a deep learning algorithm is executed, which is used to classify specific images into exacting disorders. It makes the recognition of diseases easy and discovers the treatment for the diseased plant. It finds out the pertinent tally of the pixels by evaluating the pictures with information samples. The corresponding infections are found according to the relevant calculations.

1.2 Related Work

Recently, image processing has become one of the most emerging and advanced approaches to detecting plant diseases and developing advanced technologies to improve farm productivity.

Abed and Esmaeel [3] proposed research to detect bacterial brown spots and powdery mildew infections from bean leaves. The projected method showed a precision rate of 100% for detecting diseases at the initial phases. The cluster was selected automatically for improving the performance of the launched scheme. Going into the contaminated cluster of leaves and plants was significant, as the value of clustering alterations of testing pictures was applied in the input. Therefore, the infections were identified and categorized, and highly productive and precise outcomes were attained through the projected scheme. Hossain et al. [4] proposed a support vector machine-based classification model to detect infections in plants and harvest. The investigators concluded that the projected scheme increased the discovery, fortitude, classification, and categorization competence by 93% compared to the existing classification models. This method reduced the feature withdrawal as well, which in turn reduced the dispensation time.

Khan et al. [5] described the machine vision framework, recognized the demonstration of plant infections, and scrutinized the pictures in CIELAB. The primary aim of this approach was to develop a process for discovering plant infections through the use of cascade unsupervised picture segmentation schemes. The tested outcomes demonstrated that the novel cascaded intend provided a better colour segmentation with the corroboration of contaminated areas. Mattihalli et al. [6] proposed the developments completely on the previously projected classification models utilized to discover plant infections. The automated discovery of grape leaf infection was cast in this work, which worked according to the k-means clustering technique and SVM classification model.

Bharate and Shridhonkar [7] presented a review of different techniques for discovering plant infections with the help of a picture processing method. This work included researches on plants such as apple, grapes, pepper, pomegranate, and many more. The most common and accepted classifier utilized to discover infections is the artificial neural network (ANN) classifier and support vector machine (SVM) classifier. The automated discovery of diseases resolved the difficulty of expensive areas. It was concluded that the timely discovery of infections provided help to the farmers for improving the production and Indian gross domestic production (GDP) values. Kaur et al. [8] analysed the conditions caused to the fruit yield. The study showed that the projected scheme helped detect infections in the fruits, and this method required much less time than the manual scheme. Because the noises distorted the pictures, thus the idea of denoising was demonstrated in this study as well. It was concluded that the work performed in this investigation showed that blight was a prevalent infection ailing a lot of plants and yields.

Ashourloo et al. [9] presented a range for the healthy and unaffected leaves comprising different indications, which were monitored using a picture processing spectroradiometer comprising 350–2500 nm electromagnetic area. The ground trust data sample was developed using pictures on a digital camera and computed the infections and their indications. The study mainly focused on using machine learning methods to discover and categorize plant infections. Therefore, it was concluded that the projected process was beneficial for finding and classifying disorders. Dhakate and Ingole et al. [10] studied several image processing and neural network techniques designed for performing plant disease detection and classification. In this approach, training is applied on a few images and tested for others. The images are pre-processed by applying the k-means clustering technique. The texture of images is extracted using grey level co-occurrence matrix (GLCM) and then forwarded towards the artificial neural networks (ANNs). Around 90% of accuracy is achieved as per experimental results. Satisfactory outcomes were conducted as per the results achieved after simulations. 100% accurate results have been achieved when detecting diseases from plants, around 87% by the leaf spots, approximately 85% for bacterial blight, and about 83% for fruit rot.

A brief introduction on plant disease detection using image processing and basic steps for disease detection is discussed. Moreover, the exhaustive study of critical research articles in this area, their significant contributions, suggested approaches, and obtained results are also included. In this paper, a semi-automatic system is designed to detect two diseases of soybean (glycine max) named mosaic virus and leaf spot. We used the k-means clustering approach to extract the combined colour and texture features from the diseased area of soybean leaves and classified it using the KNN algorithm. The observations are compared with the existing work to validate the efficacy of the proposed approach. Moreover, visual inspections of the leaf sample also prove the suitability of the proposed system for detection and classification.

The rest of the paper is organized as follows: Sect. 2 described the problem statement; Sect. 3 discussed the proposed plant disease detection methodology. Experimental results and analysis are discussed in Sect. 4. Finally, conclusions of the work are given in Sect. 5.

2 Problem Statement

The plant infection discovery is the method used to detect infection from the key plant leaves. The assets of the vital picture can be scrutinized into colour and texture format. The picture's colour assets symbolize the input's colour strength by utilizing red, green, and blue. The texture descriptions of the image correspond to colour illustrations of the group of pixels. The plant infection discovery procedure comprises three stages. In the primary stage, the segmentation method is executed for the segmentation of alike and unlike fraction of the entire picture. In the next step, the texture characteristics of the critical image are scrutinized. The classification method is executed to classify the picture into definite classes following their properties in the final stage. In the base document, the area relied upon segmentation is performed for the segmentation of plant leaf. The textural features are scrutinized through the GLCM algorithm, while SVM is utilized to classify infections.

A technique is presented in the next section that extracts textural and colour features for better accuracy to make plant infection research. It uses the classification technique for collecting particular plant images, detects diseases with analysis, and compares results with existing results. In that form, the steps of the disease detection process have been described in Sect. 1. We processed RGB images in the LAB colour space, and a k-means clustering algorithm was adopted for the segmentation. For feature extraction, the GLCM algorithm is applied to extract texture and colour features. Further, a support vector machine classifier will be substituted with a K-nearest neighbour classifier to increase the classification precision and compare the result with existing approaches for the images of a soybean plant.

3 Proposed Plant Disease Detection Methodology

The plant disease detection is applied to detect the disease portion from the input leaf image of the plant. The proposed technique is based on the following steps.

1. **Image Acquisition (Data Set Details)**

   The proposed algorithm developed is generic to any plant disease detection. However, for the analysis and comparison of results with the existing methods, we have used the soybean (glycine max) plant images and its two diseases named: (1) leaf spot (200 images) and (2) mosaic virus (250 images) total having 450 images captured through RGB digital cameras available at the sources named (1) IPM images [11] and (2) plant village images [12]. Sample images of two diseases are shown in Fig. 2.

All the images captured in data sets have no complex background and other noise available. They are already pre-processed and made available by the respective authority of the source.

2. **Pre-processing Phase**

   The plant leaf image given in the first step is converted to greyscale. Next, we apply the GLCM algorithm to extract the textural feature of the input image.

3. **Image Segmentation**

   The image segmentation technique is applied, which will segment the image based on their properties. The image segmentation techniques are generally applied to categorize into the region and threshold-based segmentation. In this work, the region-based k-means segmentation technique is applied for image segmentation.

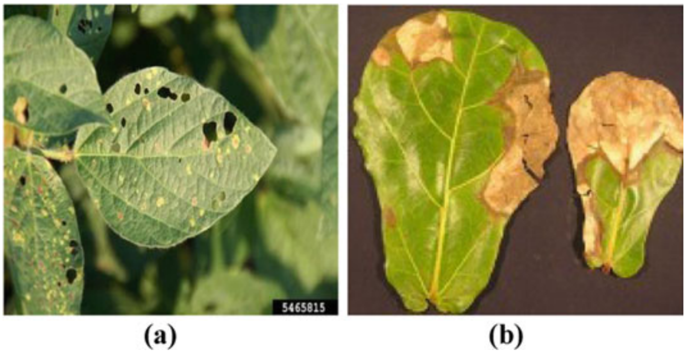


Fig. 2 **a** Leaf spot disease image, **b** mosaic virus disease image

```
Algorithm for Image Segmentation
Input: Dataset
Output: Clustered Data
Start()
   1. Read dataset and dataset has number of rows "n" and
      number of columns "m"
   2. Selection of medoid point()
   3. for (j = 0; j = m; j + +) do
   4.     Select k = data (i, j);
   5. end for
   6. Calculation of Euclidian distance()
   7. for (i = 0; i = r; i + +) do
   8.   for (j = 0; j = m; j + +)  do
   9.     A(i) = data(i);
   10.    B(i) = data(j);
   11.    Distance = $sqrt [(A(i +1) - A(i)^2) (B(j+1) - B (j)^2)]$;
   12.  end for
   13.  end for
   14.  Normalization()
   15.  for (k = 0; k = data; k + +) do
   16.      Swap k(i + 1) and k(i);
   17.  end for
   18.  Repeat from step 6 until all points get clustered.
```

After the segmentation process, any input disease leaf image is shown in Fig. 3, where the above step does three image clusters.

4. Feature Extraction

The feature plays a vital role in detecting a disease from one another. After doing segmentation, we select the collection of interest manually from the cluster. We can see more infection done by the disease, and from that selected cluster, we applied the GLCM algorithm to extract texture and colour features. We have pulled a total of thirteen features using the GLCM algorithm [13]. The colour features we have extracted are mean, standard deviation, kurtosis, skewness, and

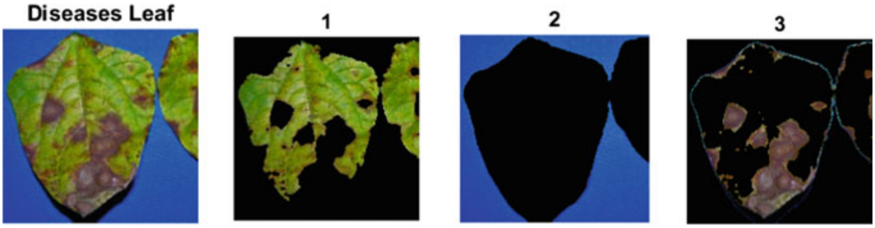


Fig. 3 Three clusters are given by k-means clustering

variance. The texture features we have extracted are contrast, co-relation, energy, homogeneity, smoothness, inverse differentiation moment, and entropy. Below is step one of the above features named "contrast" extraction using GLCM:

(i) We first obtained the number of pixels of the matrix where data is saved.
(ii) Create matrix $P[i, j]$ of the counted pixels data.
(iii) Using histogram techniques, we can check the similarities between pixels in the matrix.
(iv) The contrast factor from the matrix can be calculated as follows:

$$g = \exp\left[\frac{\text{mean}(I) - \text{minimum}(I)}{\text{maximum}(I) - \text{mean}(I)}\right]$$

(v) In last, normalize the elements of g by dividing the number of pixels.

$$g = \begin{cases} 0.8 & \text{if } g < 0.8 \\ 1.2 & \text{if } g > 1.2 \\ g & \text{otherwise} \end{cases}$$

5. **Classification of Data**

The KNN is the classification technique applied to classify similar and dissimilar data into more than one class. The learning method of k-nearest neighbour is based on the data analogy. There are n-dimensional attributes that can be represented as the training samples. The more significant part of the training samples is carried out and saved in n-dimensional pattern space. K-nearest neighbour classifier observes the pattern space for the k training samples near the unknown samples. "Closeness" is defined in terms of Euclidean distance. Like decision tree induction and backpropagation, nearest neighbour classifiers assigned break even with weight to every attribute. This may bring about confusion when there are numerous irrelevant attributes in the data. Nearest neighbour classifiers can likewise be utilized for prediction, that is, to give back a genuinely valued prediction for a given unknown sample. For this situation, the classifier gives back the average value of the genuine value associated with the unknown sample's k-nearest neighbours. The k-nearest neighbours' algorithm is among the simplest of all machine learning algorithms and requires three things:

(1) Feature space (training data).
(2) Distance metric: to compute the distance between records.
(3) The value of k: The number of nearest neighbours to retrieve from which to get the majority class.

Figure 4 shows a complete flow chart of the proposed approach.

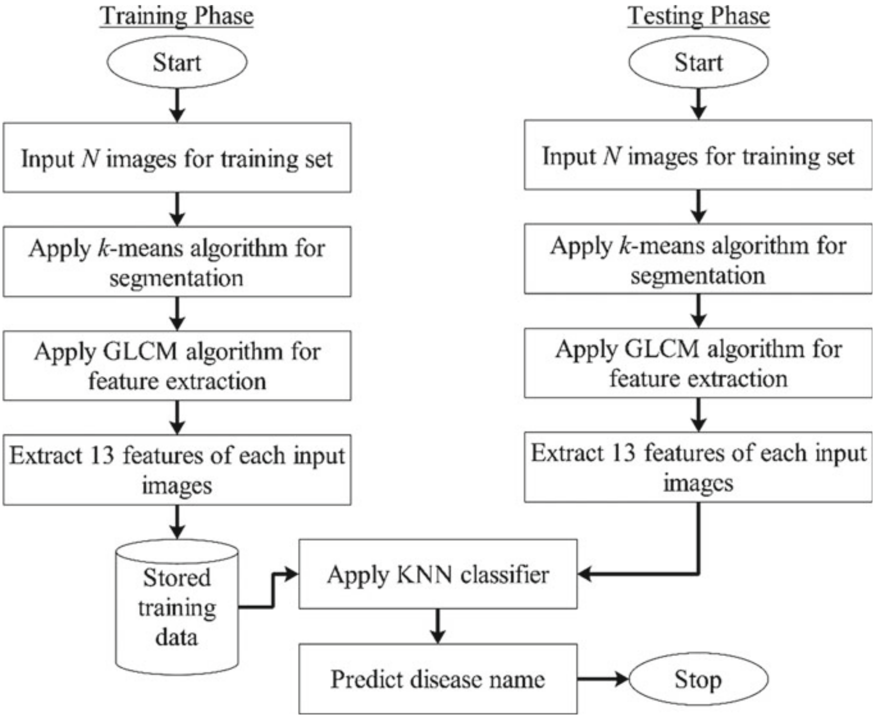

Training Phase
Start
Input *N* images for training set
Apply *k*-means algorithm for segmentation
Apply GLCM algorithm for feature extraction
Extract 13 features of each input images
Stored training data
Testing Phase
Start
Input *N* images for training set
Apply *k*-means algorithm for segmentation
Apply GLCM algorithm for feature extraction
Extract 13 features of each input images
Apply KNN classifier
Predict disease name
Stop

Fig. 4 Flow chart of the proposed approach

4 Experimental Results and Analysis

For numerical analysis, we used the MATLAB 2017 software with an image processing toolbox. For the testing purpose of the proposed approach, the K-fold cross-validation technique is adopted, and the overall accuracy of the proposed work is computed. Further, we compared the result with other existing techniques' accuracy. We used the sample of 450 images of soybean leaves having two diseases named mosaic virus (250 Images) and leaf spot (200 Images).

A. K-Fold Cross-Validation Method

It is a re-sampling method used to assess the machine learning algorithm when limited data is available. This technique consists only of a single parameter, K, where the given information is distributed into K groups. If we choose $K = 3$, it intends three cross-fold validations as the reference in the model.

This cross-validation method is helpful in the estimation of the ability of the machine learning model on unobserved data. This approach uses a limited sample for estimating the model's behaviour that can be utilized to estimate the characteristics of the data that is not accessed during the model's training. The step-by-step approach is given as follows:

1. First, data is shuffled randomly.
2. An available data set is distributed in the K group ($K = 3$).
3. Choose each group one by one and
    a. assign that group as a test data set or a hold-out.
    b. remaining groups as a training data set.
    c. fit the given model on the training set and then evaluate it on the test set.
4. Keep the estimation score and then remove the model.
5. In last, based on the estimation score, the behavioural characteristic of the model can be obtained.

We have prepared a confusion matrix and a tale often used to show the performance of the classification model on test data for those whose actual values are available. Table 1 shows the confusion matrix for different samples of mosaic virus and leaf spot.

From Table 1, one can observe that how often the classifier gives the correct result and here, in each fold ($K = 3$) of 150 images, how many times was disease correctly classified out of 150 prints. Hence, the average accuracy is computed at 92%. Table 2 shows various approaches applied for soybean disease detection and methods used for classification based on the multiple features, classifiers, and several images used as data sets to observe the result.

In Fig. 5, the accuracy of the proposed approach is compared with the other state-of-the-art methods [8, 14–16], and we observed 92% accuracy in disease classification using tested samples of soybean leaves using the proposed method. The other approaches shown in Fig. 5 are also tested on the same soybean plant leaves images, but different diseases and images are used differently from the proposed

Table 1 Confusion matrix for samples of mosaic virus and leaf spot

	Predicted result	
Total = 450	Mosaic virus	Leaf spot
Mosaic virus	236	14
Leaf spot	22	178

Table 2 Proposed work with existing approaches

Reference paper	Features	Classifier	Number of images
Shrivastav et al. [14]	Shape	KNN	57
Dondawate and Kokare [15]	Texture	LDA	120
Shrivastav et al. [16]	Shape	SVM	100
Kaur et al. [8]	Colour + gabour and Texture + colour	SVM	539
Proposed work	Texture + colour	KNN	450

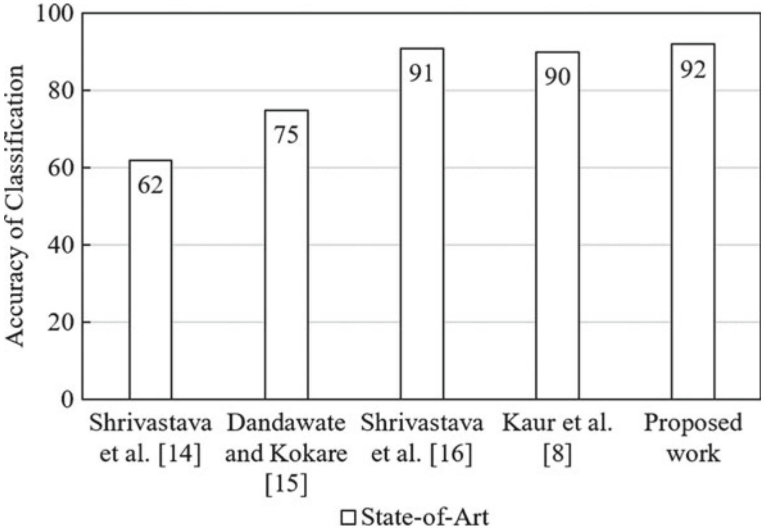


Fig. 5 Accuracy comparison of the proposed method with existing approaches

work. Specifically, comparing the proposed work with the recent study in [8], where an SVM classifier is used, we observed 2% accuracy in classification.

Challenges

Below are some critical and considerable plant infection detection research areas showing that many other areas can be open to research [7].

1. Presence of complex backgrounds that cannot be easily separated from the region of interest.
2. Boundaries of the symptoms are not well defined.
3. Uncontrolled capture conditions present characteristics which make the image analysis more challenging.
4. The disease may possess very different characteristics depending on its stage of development and sometimes on where it is located on the plant.
5. In this area of research, there is a lack of available images to use for disease detection.
6. Most studies done in this area have chosen to separate only diseases with relatively different symptoms.
7. Mostly, it is assumed that only one disease is present in each image, but other diseases and other disorders such as nutritional deficiency and pests might manifest simultaneously.
8. Different stages of a particular disease may produce other symptoms.
9. Humidity, exposure to sunlight, temperature, and wind factors may alter the symptoms of the disease.

5 Conclusion and Future Work

The proposed research study demonstrates that plant infection discovery requires three significant phases: characteristic retrieval, segmentation, and classification. The accessible method utilizes the GLCM algorithm for the extraction of texture descriptions. The segmentation of input pictures is performed through the application of the k-means clustering algorithm. The texture descriptions of the input picture are retrieved with the help of the GLCM algorithm. The GLCM algorithm is used for the extraction of thirteen reports of the image. The classification algorithm KNN is used to compare with the other existing classification algorithm like SVM. The MATLAB tool is used for the execution of projected and accessible algorithms. The outcomes of the projected methodology are examined utilizing accuracy. It is scrutinized that the projected method gives better performance than existing schemes using described constraint. Also, some other comparison is made with the proposed work result and tried to observe where and how the accurate result came to compare to different existing approaches in this area of disease discovery.

This work can be extended by using any one of the following points:

1. The proposed algorithm can be tested on several other data samples for analysing their recital.
2. The proposed algorithm can be tested, making it available on a mobile phone application developed and taking real-time images on the farm.
3. Like projected algorithms, we can apply different not applied methods in each disease detection workflow and compare them with existing ones.
4. We can define and prepare some index that shows the disease's particular stage, including detecting the disease process.
5. We can develop systems that can adequately differentiate between an infection and a deficiency, which is another existing research topic.

References

1. Bashir S, Sharma N (2012) Remote area plant disease detection using image processing. IOSR J Electron Commun Eng 2(6):31–34
2. Ghaiwat SN, Arora P (2014) Detection and classification of plant leaf diseases using image processing techniques: a review. Int J Rec Adv Eng Technol 2(3):1–7
3. Abed S, Esmaeel AA (2018) A novel approach to classify and detect bean diseases based on image processing. In: IEEE symposium on computer applications and industrial electronics (ISCAIE 2018), pp 297–302
4. Hossain MS, Mou RM, Hasan MM, Chakraborty S, Razzak MA (2018) Recognition and detection of tea leaf's diseases using support vector machine. In: 14th international colloquium on signal processing and its applications (CSPA 2018), pp 150–154
5. Khan ZU, Akram T, Naqvi SR, Haider SA, Kamran M, Muhammad N (2018) Automatic detection of plant diseases; utilizing an unsupervised cascaded design. In: 15th international bhurban conference on applied sciences and technology (IBCAST 2018), pp 339–346

6. Mattihalli C, Gedefaye E, Endalamaw F, Necho A (2018) Real time automation of agriculture land, by automatically detecting plant leaf diseases and auto medicine. In: 32nd international conference on advanced information networking and applications workshops (WAINA 2018), pp 325–330
7. Bharate AA, Shirdhonkar M (2017) A review on plant disease detection using image processing. In: International conference on intelligent sustainable systems (ICISS 2017), pp 103–109
8. Kaur S, Pandey S, Goel S (2018) Semi-automatic leaf disease detection and classification system for soybean culture. IET Image Proc 12(6):1038–1048
9. Ashourloo D, Aghighi F, Matkan AA, Mobasheri MR, Rad AM (2016) An investigation into machine learning regression techniques for the leaf rust disease detection using hyperspectral measurement. IEEE J Select Top Appl Earth Observ Remote Sens 9(9):4344–4351
10. Dhakate M, Ingole A (2015) Diagnosis of pomegranate plant diseases using neural network. In: 5th National conference on computer vision, pattern recognition, image processing and graphics (NCVPRIPG 2015), pp 1–4
11. IPM images. Accessed on 15 Nov 2021 [Online]. Available: www.ipmimages.org
12. Plant village images. Accessed on 15 Nov 2021 [Online]. Available: www.plantvillage.com
13. Hu Y, Zhao C, Wang H (2008) Directional analysis of texture images using gray level co-occurrence matrix. In: Asia-Pacific workshop on computational intelligence and industrial application, vol 2, pp 277–281
14. Shrivastava S, Singh SK, Hooda DS (2015) Color sensing and image processing-based automatic soybean plant foliar disease severity detection and estimation. Multimedia Tools Appl 74(24):11467–11484
15. Dandawate Y, Kokare R (2015) An automated approach for classification of plant diseases towards development of futuristic decision support system in indian perspective. In: International conference on advances in computing, communications and informatics (ICACCI 2015), pp 794–799
16. Shrivastava S, Singh SK, Hooda DS (2017) Soybean plant foliar disease detection using image retrieval approaches. Multimedia Tools Appl 76(24):26647–26674

Comparison of BERT-Base and GPT-3 for Marathi Text Classification

Chandrashekhar S. Pawar and Ashwin Makwana

Abstract In text summarization process, the text classification task plays an important role. So, to classify natural language text or text documents, various machine learning techniques are available. Within this paper, we have checked the performance of BERT and GPT-3 on the Marathi Polarity Labeled Corpora (MPLC) a tourism dataset. The MPLC dataset consists of 75 positive and the same number of negative reviews. The main purpose of this paper is to put GPT-3 into the limelight with its useful features. For this purpose, we compared GPT-3 with the best performing BERT model, and finally, in the race of evaluation, the GPT-3 model beats the BERT-base model.

Keywords Pre-trained models (PTMs) · Text summarization · Open generative pre-trained transformer (GPT) · Text classification · BERT

1 Introduction

In NLP, text classification is used as a basic task in various applications like text summarization, topic labeling, sentiment analysis, intent-detection, etc. [1]. In text summarization, text classification is required when we want categorize positive and negative reviews for recommending the product (whether it is good to buy or not). Similarly, in multi-document summarization, text classification is helping us to obtain required information from multi-source documents on one subject [2].

C. S. Pawar (✉)
Department of Computer Science & Engineering, Devang Patel Institute of Advance Technology and Research, Faculty of Technology and Engineering (FTE), Charotar University of Science and Technology, Changa, Gujarat 388421, India
e-mail: chandrashekharpawar.dcs@charusat.ac.in

A. Makwana
Department of Computer Engineering, Chandubhai S Patel Institute of Technology, Faculty of Technology and Engineering (FTE), Charotar University of Science and Technology, Changa, Gujarat 388421, India
e-mail: ashwinmakwana.ce@charusat.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_40

For the text classification, tasks like classification of topics, news and for document classification various approaches are used and these approaches can be categorized in two categories, first rule-based and, second one is machine learning-based methods [3].

In rule-based methods, we must have detail domain knowledge to set the predefined rules, whereas in machine learning methods learn how to classify the data based on pre-labeled data and also from unlabeled data by finding relation between words.

In deep learning, rapid development has been done in past few years, with that various model parameters are introduced. These model parameters require huge datasets to train and also to avoid overfitting problem. But the task to prepare large datasets with labels is a challenging one in many NLP solutions. So nowadays, researchers prefer to use unlabeled large dataset. It results in increasing the performance of NLP tasks by using the representation provided by pre-trained models on large unlabeled dataset [4].

Within the study, we found that pre-trained language models (PTMs) provide much support in learning and understanding natural languages representations with large unlabeled data [4]. For text classification, the PTMs such as bidirectional encoder representation from transformer (BERT) [5], ELMo [6] and OpenAI generative pre-trained transformer (GPT) model [7] provided good result. But in February 2019, OpenAI released the GPT-2 [8] and in July 2020, GPT-3 [9] is a language model which is empowered by neural network.

Within the paper, we consider the only two PTMs, BERT and GPT-3 for text classification on Marathi Polarity Labeled Corpora (MPLC) a tourism dataset. First in Sect. 2, we discuss about MPLC dataset. BERT model for text classification is in Sect. 3. Next in Sect. 4, we understand GPT-3 model and its use in text classification. After knowing about both BERT and GPT-3, we have described the advantage of GPT-3 over BERT, within Sect. 5. And in the last, we conclude our work with the benefits of GPT-3 in text summarization.

2 Literature Review of PTMs

Pre-training is a successful method for learning the features in deep neural networks, and it is need to be fine-tuned on subsequent processes. Deep learning had its triumph in 2006, when pre-trained greedy layer-wise approach (which is unsupervised one) was coupled with the supervised fine-tuning [10].

2.1 PTMs for Word Embedding

The use of dense vectors to represent words has a long history [11]. In the pioneering work of neural network language model (NNLM) [12], the term “modern” word embedding is used for the first time. Reference [13] demonstrated that using word embedding (which is pre-trained) with unlabeled data might enhance several NLP jobs greatly.

Rather than language modeling, to reduce computational complexity, they use word embedding which is trained with the mathed set of rating job. Their study was the one to try to extract general word embeddings from unlabeled data for use in other applications. Deep neural networks are not required to create good word embeddings, as shown in Ref. [14].

Shallow architectures include the continuous Bag-of-Words (CBOW) and Skip-Gram (SG) models. They are capable of learning elevated word embedding that grasp underlying syntactic and semantic links between words, despite their simplicity. Word2vec is a popular implementation of these models that provides previously trained word embedding for a wide range of NLP workloads. GloVe [15] is also a widely used method for generating previously trained word embedding from a large corpus utilizing globe word-word co-occurrence data.

Despite the fact that previously trained word embedding have proven to be extremely effective in NLP applications, they are context-insensitive and are often learnt using shallow models. Used on a subsequent assignment, all remaining model must be trained from scratch.

During the same time period, many studies sought to learn paragraph, phrase or document embedding, such as paragraph vector [16], Skip-thought vectors [17] and Context2Vec [18]. These sentence embedding methods, unlike their modern successors, attempt to encode input sentences into a fixed-dimensional vector form rather than the contextual form for each word.

2.2 PTMs for Contextual Encoders

Because many NLP tasks go beyond the word level, pre-training the neural encoders at the sentence or larger range makes sense. The output vectors of neural encoders are also called as contextual word embedding since they communicate the semantics of words based on their context.

In Ref. [19], the first useful example of PTM for NLP was given. Pre-training LSTMs with a language model (LM) or a sequence autoencoder can improve LSTM training and generalization in a range of text classification tasks, according to the authors.

Reference [20] encoder pre-trained a shared LSTM with LM and fine-tuned it for a multi-task learning framework (MTL). They figured out how to do pre-training and fine-tuning. By modifying MTL, you can boost its efficiency even more. A number of different text classification tasks like Seq2Seqare identified in Ref. [21]. Unsupervised pre-processing has the potential to significantly improve model training. The model was fine-tuned using labeled data after initializing the encoder and decoder weights with pre-trained weights from two language models. Aside from pre-training, LM pre-trained a deep neural network in [22] the contextual encoder.

Machine translation is used to create an attentional sequence-to-sequence LSTM encoder model.

OpenAI GPT [23] and BERT [5] are two notable examples of deep PTMs that have demonstrated their ability to acquire universal language models. A growing variety of self-supervised tasks is suggested in addition to LM to help PTMs capture more knowledge from huge textual data. Fine-tuning is now the standard method for adapting PTMs for downstream activities since ULMFiT and BERT.

3 Dataset

The Marathi Polarity Labeled Corpora (MPLC) a tourism dataset [24] is used for the text classification experimentation. The MPLC dataset consists of 75 positive reviews and 75 negative reviews in Marathi language only. In our research, we focus on Marathi language text classification. The most of datasets available over the Internet are multilingual. So, here in text classification experimentation, MPLC dataset is used to verify the accuracy of classifying positive and negative review. Following Fig. 1 shows the example of positive review from MPLC dataset.

Within this study, we considered 200 words limit in one review and sentences are formed using 20 words or less than 20 words and at the most 6 sentences in one review. Figure 2 shows the one of negative review from MPLC dataset.

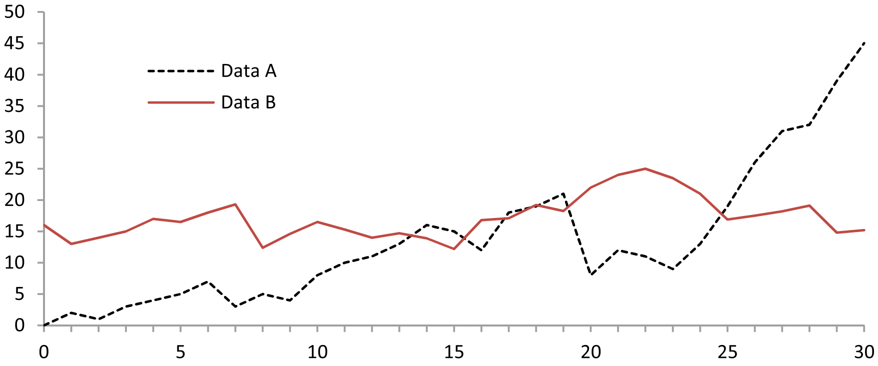


Fig. 1 Example of positive review from MPLC dataset

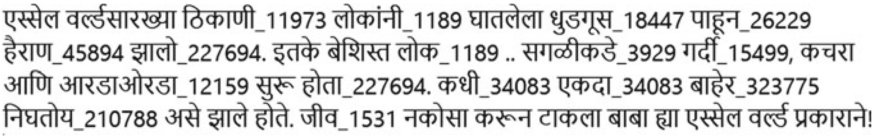

एस्सेल वर्ल्डसारख्या ठिकाणी_11973 लोकांनी_1189 घातलेला धुडगूस_18447 पाहून_26229
हैराण_45894 झालो_227694. इतके बेशिस्त लोक_1189 .. सगळीकडे_3929 गर्दी_15499, कचरा
आणि आरडाओरडा_12159 सुरू होता_227694. कधी_34083 एकदा_34083 बाहेर_323775
निघतोय_210788 असे झाले होते. जीव_1531 नकोसा करून टाकला बाबा ह्या एस्सेल वर्ल्ड प्रकाराने!

Fig. 2 Example of negative review from MPLC dataset

4 Performance Evaluation Metrics

For evaluation, True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) are defined as below:

- **True Positive**: It is a scenario where the review is classified as a positive review and the actually it is a positive review.
- **True Negative**: It is a scenario where the review is classified as a negative review and the actually it is a negative review.
- **False Positive**: It is a scenario where the review is classified as a positive review but the actually it is a negative review.
- **False Negative**: It is a scenario where the review is classified as a negative review but the actually it is a positive review.

A confusion matrix is a for visual representation of a classifier's performance, and a confusion matrix is used which is based on the four values described above (TP, FP, TN, FN). A confusion matrix is formed by plotting these against one other, as show in Fig. 3.

The performance evaluation metrics are precision, recall and F-score, calculated as shown below:

- **Precision**:

$$\text{Precision} = \frac{\text{True Positive (TP)}}{\text{True Positive (TP)} + \text{False Positive(FP)}} \tag{1}$$

In our case, it would look like this:

$$\text{Precision} = \frac{\substack{\text{Actual: Positive Review}\\ \text{Classified As:Positive Review}}}{\substack{\text{Actual: Positive Review}\\ \text{Classified As:Positive Review}} + \substack{\text{Actual: Negative Review}\\ \text{Classified As:Positive Review}}} \tag{2}$$

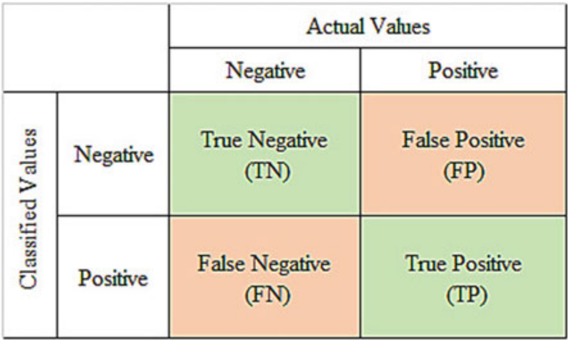


Fig. 3 Confusion matrix

		Actual Values	
		Negative	Positive
Classified Values	Negative	True Negative (TN)	False Positive (FP)
	Positive	False Negative (FN)	True Positive (TP)

- **Recall**:

$$\text{Recall} = \frac{\text{True Positive (TP)}}{\text{True Positive (TP)} + \text{False Negative (FN)}} \tag{3}$$

In our case, it would look like this:

$$\text{Recall} = \frac{\substack{\text{Actual: Positive Review} \\ \text{Classified As:Positive Review}}}{\substack{\text{Actual: Positive Review} \\ \text{Classified As:Positive Review}} + \substack{\text{Actual: Positive Review} \\ \text{Classified As:Negative Review}}} \tag{4}$$

- **F1-score**:

$$F1 - \text{score} = 2\frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5}$$

5 BERT for Text Classification

In this section, first we describe the basics about BERT and after that we discuss about the BERT implementation for text classification.

5.1 BERT Model

BERT helps to remove unidirectional constrains by mask language model with the objective of pre-training, encouraged by [5]. The BERT-base model is use for text classification. In the used BERT-base model, we denote number of transformers as TB and TB = 12 (i.e., number of layers), 12 self-attention heads are denoted by A = 12, and hidden size is 786 represented by H (i.e., H = 786).

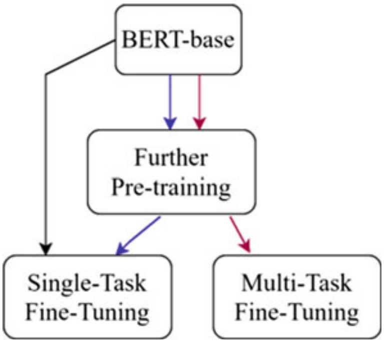
5.2 Methodology

The text classification is achieved through BERT-base model by two step process: In first step, pre-train the BERT, and next in second step, BERT is fine-tuned. The methods of fine-tuning BERT are shown in following Fig. 4, as like in [25, 26].

The syntactic and semantic data at different level can be captured by neural network at different level [27, 28].

When we are using BERT for specific task, overfitting issue must be taken into account. The problem of overfitting may occur when BERT model is train with small size dataset. So, to avoid overfitting problem, we can use pre-trained BERT model

Fig. 4 Different methods of fine-tuning BERT



which is already train on large size dataset. And this BERT model is fine-tuned by training the model with few sample.

It is good to use good optimizer along with applicable learning rate. The optimizers work is to help the neural network to adjust learning rate and weights for minimizing the losses. We use Adam optimizer [29] as too fast, and it is able to converge rapidly and it rectifies vanishing learning rate and high variance. One can easily know that the general category information is found in the BERT's lower level and one can fine-tuned them using diverse learning rates. As done in [27], the parameters θ split into $\{ \theta^1, \ldots, \theta^n \}$, within this nth BERT layer parameters are contain in θ^n . By using following formula, we updated the parameters:

$$\theta_t^n = \theta_{t-1}^n - \eta^n . \nabla_{\theta^n} J (\theta) \tag{6}$$

where, η^n is a learning rate for nth layer.

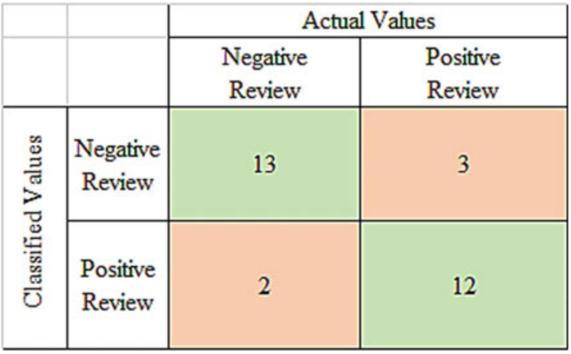
5.3 *Results*

With BERT-base, we used 80% of MPLC dataset for training purpose and 20 % of MPLC dataset for testing purpose. The evaluation metrics are precision, recall and F-score, calculated as per the confusion matrix shown in Fig. 5.

And the evaluation results are given in Table 1.

6 GPT-3 Model and Its Use in Text Classification

In this section, first we describe the basics about GPT-3 and after that we discuss about the use of GPT-3 for text classification.



		Actual Values	
		Negative Review	Positive Review
Classified Values	Negative Review	13	3
	Positive Review	2	12

Fig. 5 Confusion matrix for BERT-base model

Table 1 BERT-base evaluation results

	Precision	Recall	*F*-score
BERT-base	0.80	0.85	0.82

6.1 GPT-3 Model

In past few years, GPT has attracted the attention of researchers working in NLP domain. The newly introduced model is GPT-3, trained on 175 billion parameters. Now, we do not require fine-tuning or gradient update in GPT-3. The GPT-3 was publicized in [9].

The important factors about GPT-3 are listed below:

- Models: It has 8 diverse models which ranges from 125 million to 175 billion parameters.
- Architecture: It is an autoregressive model and with the architecture of decoder only.
- Learning: Few shots are required for learning purpose. It does not require fine-tuning or gradient update.

6.2 Methodology

Three methods are accepted by GPT-3 for learning. So, large datasets are not required for extrapolation on new problems. Instead, it can learn from no data (zero shot learning), only 01 sample (one shot learning) or limited samples (few shot learning) [9].

Here, we used few shot learning method with GPT-3 for text classification. For few shot learning, we considered only 20% of dataset and testing was done on

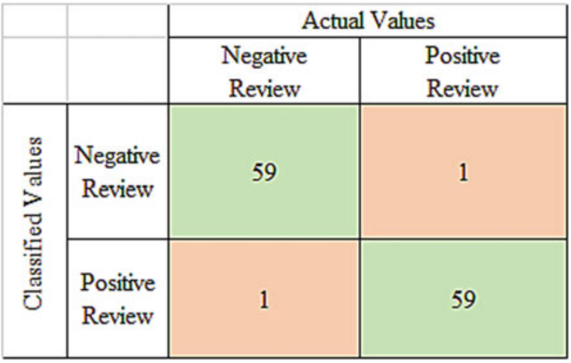
Table 2 GPT-3 small-architecture, learning rate and batch size

Model Name	n_{layers}	n_{heads}	d_{heads}	d_{model}	n_{params}	Learning rate	Batch size
GPT-3 small	12	12	64	768	125M	6.0×10^{-4}	0.5 M

remaining 80% of dataset. Within this paper, we have used GPT-3 small model for text classification task. The batch size, learning parameter and architecture for GPT-3 small model are given in Table 2, as per [9].

6.3 *Results*

For GPT-3 small model evaluation, we have used the same evaluation metrics, i.e., precision, recall and F-score. And for calculating these metrics, we have used the values shown in following Fig. 6 confusion matrix for GPT-3 small. And the evaluation results are given in Table 3.



		Actual Values	
		Negative Review	Positive Review
Classified Values	Negative Review	59	1
	Positive Review	1	59

Fig. 6 Confusion matrix for GPT-3 small model

Table 3 GPT-3 small-evaluation results

	Precision	Recall	F-score
GPT-3 small	0.98	0.98	0.98

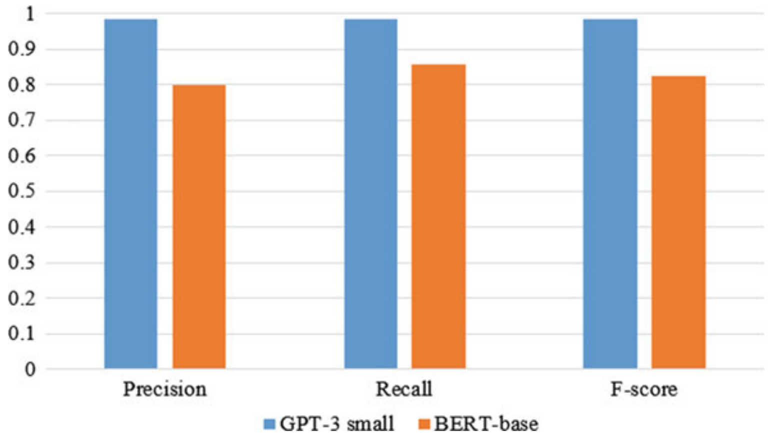


Fig. 7 Comparison of evaluation metrics for GPT-3 small and BERT-base models

7 Comparison of GPT-3 and BERT

Tables 2 and 3 give the evaluation results for BERT-base and GPT-3 small, respectively. By observing the obtain result values, we conclude that GPT-3 small model performed better than BERT-base model.

And for this, GPT-3 used only 20% of dataset for training, it means with less training time, we can obtain good results.

As we mentioned in above in Sect. 4, the added advantage of GPT-3 is that it does not required fine-tuning or gradient update. The graphical representation for comparison of evaluation metrics is depicted in Fig. 7.

8 What Distinguishes GPT-3 from BERT?

Things which stand out from for GPT-3 are:

- The most notable attribute of the GPT-3 is its size. It is over 470 times greater in size BERT model.
- In terms of architecture, BERT has the upper hand. It challenges that have been trained on that are better at capturing the latent relationship between text in various problem situations.
- The GPT-3 learning approach is straightforward and can be used to solve a variety of issues in which sufficient data is lacking. As a result, as compared to BERT, GPT-3 should have a broader use.

9 Conclusions

When Google first announced BERT, it generated a lot of buzz; however, the buzz around the GPT-3 model has entirely overshadowed BERT's potential. Much of this is due to the fact that, unlike BERT, OpenAI's GPT-3 does not necessitate a large amount of data for training.

GPT-3 shows that a language model with adequate training data can solve NLP task it has never seen before. GPT-3, in other words, investigates the model as a broad solution for a variety of downstream applications without requiring fine-tuning. It can perform machine translation, question-answering, conceptual task reading, poetry scripting and simple math.

References

1. Yang J, Bai L, Guo Y (2020) A survey of text classification models, pp 327–334
2. Zhang P, Li C (2009) Automatic text summarization based on sentences clustering and extraction, pp 167–170
3. Minaee S, Kalchbrenner N, Cambria E, Nikzad N, Chenaghlu M, Gao J (2021) Deep learning based text classification: a comprehensive review. https://dl.acm.org/doi/10.1145/3439726
4. Qiu X, Sun T, Xu Y, Shao Y, Dai N, Huang X (2020) Pre-trained models for natural language processing: a survey, pp 1–26
5. Devlin J, Chang M-W, Lee K, Toutanova K (2019) BERT: pre-training of deep bidirectional transformers for language understanding, pp 4171–4186
6. Peters ME, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, Zettlemoyer L (2018) Deep contextualized word representations, pp 2227–2237
7. Radford A, Narasimhan K, Salimans T, Sutskever I (2018) Improving language understanding by generative pre-training. OpenAI Blog, pp 399–438
8. Radford A, Wu J, Child R, Luan D, Amodei D, Sutskever I (2019) Language models are unsupervised multitask learners. OpenAI blog 1(8):9
9. Brown TB, Mann B, Ryder N, Subbiah M, Kaplan J, Dhariwal P, Neelakantan A, Shyam P, Sastry G, Askell A et al (2020) Language models are few-shot learners. arXiv preprint arXiv:2005.14165
10. Hinton GE, Salakhutdinov RR (2006) Reducing the dimensionality of data with neural networks. Science 313(5786):504–507
11. Hinton GE (1984) Distributed representations
12. Bengio Y, Ducharme R, Vincent P, Janvin C (2003) A neural probabilistic language model. J Mach Learn Res 3:1137–1155
13. Collobert R, Weston J, Bottou L, Karlen M, Kavukcuoglu K, Kuksa P (2011) Natural language processing (almost) from scratch. J Mach Learn Res 12(ARTICLE):2493–2537
14. Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013) Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems, pp 3111–3119
15. Pennington J, Socher R, Manning CD (2014) Glove: global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp 1532–1543
16. Le Q, Mikolov T (2014) Distributed representations of sentences and documents. In: International conference on machine learning, pp 1188–1196
17. Kiros R, Zhu Y, Salakhutdinov RR, Zemel R, Urtasun R, Torralba A, Fidler S (2015) Skip-thought vectors. In: Advances in neural information processing systems, pp 3294–3302

18. Melamud O, Goldberger J, Dagan I (2016) context2vec: learning generic context embedding with bidirectional LSTM. In: Proceedings of the 20th SIGNLL conference on computational natural language learning, pp 51–61
19. Dai AM, Le QV (2015) Semi-supervised sequence learning. Adv Neural Inform Process Syst 28:3079–3087
20. Liu P, Qiu X, Huang X (2016) Recurrent neural network for text classification with multi-task learning. In: Proceedings of the international joint conference on artificial intelligence, New York, pp 2873–2879
21. Ramachandran P, Liu PJ, Le QV (2016) Unsupervised pretraining for sequence to sequence learning. In: Proceedings of the conference on empirical methods in natural language processing, Copenhagen, pp 383–391
22. McCann B, Bradbury J, Xiong C, Socher R (2017) Learned in translation: contextualized word vectors. In: Proceedings of the advances in neural information processing system, Long Beach, pp 6294–6305
23. Radford A, Narasimhan K, Salimans T, Sutskever I (2018) Improving language understanding by generative pre-training
24. Balamurali A, Joshi A, Bhattacharyya P (2012) Cross-lingual sentiment analysis for Indian languages using linked wordnets, pp 73–82
25. Taylor WL (1953) "cloze procedure": a new tool for measuring readability. J Q 30(4):415–433
26. Sun C, Qiu X, Xu Y, Huang X (2019) How to fine-tune BERT for text classification? pp 194–206
27. Howard J, Ruder S (2018) Universal language model fine-tuning for text classification. arXiv preprint arXiv:1801.06146
28. Yosinski J, Clune J, Bengio Y, Lipson H (2014) How transferable are features in deep neural networks? arXiv preprint arXiv:1411.1792
29. Kingma DP, Ba JL (2015) Adam: a method for stochastic optimization 3rd international conference on learning representations, ICLR 2015, San Diego, CA, USA, May 7–9, 2015, conference track proceedings

RDF Query Processing: Relational Vs. Graph Approach

Ami Pandat and Minal Bhise

Abstract The data volume for the resource description framework (RDF) is growing rapidly. To query this large amount of data, two types of query processing approaches are there: the relational approach and graph-based approach. The relational approach offers better scalability and good security, but it suffers from query joins. The graph-based approach helps to eliminate query joins. Both relational and graph-based approach have their own advantages. This paper presents the performance analysis of relational database management system (RDBMS) and graph database management system (GDBMS) in terms of query processing. The evaluation of query execution time (QET) is carried out for the same benchmark linked observation data (LOD). A detailed comparison is carried out using two popular graph-based tools, Neo4j, and DGraph with relational database PostgreSQL. Neo4j uses cipher-query language, and DGraph uses GraphQL+−, and PostgreSQL uses standard SQL language for the RDBMS. The experiments are performed for four different types of queries. It has been observed that RDBMS outperforms GDBMS for star and range queries, and graph database works well with join and projection queries. We have extended our evaluation for the comparison of available graph tools, and it has been found that DGraph outperforms Neo4j for all the types of queries and works 2.5% faster for the query execution time as well.

Keywords Cypher · Graph database · GraphQL+− · RDF · SQL

1 Introduction

The resource description framework (RDF) [13] is standardized with W3C and highly used for data interchange and data exchange for semantic Web resources. Though RDBMS is a traditional database management approach, to analyze interactive rela-

A. Pandat (✉) · M. Bhise
Distributed Databases Group, DAIICT, Gandhinagar, India
e-mail: ami_pandat@daiict.ac.in

M. Bhise
e-mail: minal_bhise@daiict.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*, Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_41

tionships of modern day RDF applications, graph is an appropriate data structure. RDBMS is still a mainstream database but since last few years, graph-based applications like social network, fraud detection, purchase servicing gained attention, and lead big ata processing at a certain level of attraction. The graph-based technique suits up well with the RDF data as RDF and graph both can be represented in the form of a triple. RDF comprises of ⟨subject, predicate, and object⟩, and graph comprises of ⟨Vertices, edges, and relationship between them⟩. Graph-based management has an advantage that it maintains the original structure of the data, and it eliminates join operation as it traverses all the data nodes through the edges in a graph. To query RDF data using this approach, it uses a query graph for the same. The SPARQL query language is designed for basic RDF query execution. It works using a basic graph pattern (BGP) [18] which is defined by the collection of triples. The basic condition of this query language is designed by variables.

The real life, semantic Web-based applications for RDF data use various sources on the Web. It could be transportation data, sensor data, weather information data, educational institutes data, etc. As the usage of these applications increases, amount of RDF data has also been marked proliferated over the years. Web data management of this proliferated data is gaining attention. To manage all these applications with large volume of data, a graph-based system and relational system are two approaches.

Both RDBMS and GDBMS have their own query processing frameworks. The relational system deals with SQL—structured query language whereas graph database system deals with NoSQL query languages. NoSQL is referred to as the non-SQL or non-relational database. This is a kind of database that uses a representation of data other than tabular relations. Due to this property, these types of databases are generally used in real-time Web applications for the big data, and their usage is increasing over the time. RDBMS has standard query plan whereas for GDBMS, each and every graph technology has their own query languages and structure [2]. Query processing includes the format of input data, query plan, output format, and the data that will be analyzed or searched for the query. The RDF data either could be used as .csv file or .ttl file. As RDF is compatible with semantic Web, the tripled formed data are able to specify semantics, too. The semantic compatible data will be displayed as turtle file.

This work discusses an analysis of two types of the query execution plan for relational and graph-based systems. Both graph-based tools, Neo4j and DGraph, are compared with traditional RDBMS. Further, paper is organized as follows: Sect. 2 describes survey taken for the related work. Query execution plan and methodology of the tools are described in Sect. 3. Section 4 discusses the environmental setup and details of the dataset and query set. Results of the experiments are discussed in Sect. 5. The last section concludes the paper.

2 Related Work

There has been marked exponential growth in the use of semantic Web applications and so as RDF data is increasing. The management of RDF data has been a matter of contention in semantic Web and related research areas. It can be addressed by two techniques: relational and graph-based approach [8]. As discussed in [18], there are three graph-based systems to manage RDF data which are as follows: gStore, gStore-D, and gAnswer. gStore and gStore-D are designed to support efficient SPARQL query evaluation, while gAnswer is designed for users to make them feel easier to access RDF repository. To make query processing efficient, reducing query execution time (QET) is the main objective. Data fragmentation, summarization, and skipping are some of the ways to deal with the same. The fragmentation and allocation of a graph using proper query processing can result in efficient RDF data management [11]. DWAHP [9] and LBSD [10] are the two approach for data partitioning. The former does the partitioning for relational data, and later works with graph-based RDF fragmentation using Neo4j [3, 6]. A novel vertical partitioning method that works well with reliability is discussed in [1]. SIVP [15] is a hybrid approach for structural indexing and vertical partitioning and shows 34% gain over vertical partitioning approach.

A recent survey [14] for the graph technologies discusses various applications implemented using graph database, especially with RDF storage. It includes: risk analysis and compliance in the field of finance and health, fraud detection (e.g., detecting cybercrime), and social network analysis (e.g., finding out most viral users). The authors have interviewed the developers of these real-life applications, and it has been found that main system of record for all such applications is RDBMS; GDBMS and RDF are used as part of implementation. Hence, both relational and graph approach have their own usages, and there is no borderline for the same.

Both relational and graph-based approach have their own storage and retrieval methodologies. After storing the data, query processing framework plays a vital role to query the data. Relational database uses structured query language (SQL), a standard language for the relational query processing. Graph-based query languages are mostly used when are dealing with NoSQL databases. NoSQL databases are those which does not have any tabular form, so they are not having particular attributes to select.

2.1 Relational and Graph-Based Query Processing Tools

Relational database has been mainstream for the traditional database management system, and PostgreSQL [12] has been found one of the most appropriate tool for the research and implementation. This work also has been carried out using PostgreSQL only as a RDBMS representative.

For the experimental setup, we have explored and searched for different tools and technologies available for query processing. One of the graph tool Orient DB is a distributed graph tool, but this tool requires manual setup for edge creation. Tiger graph is faster than Neo4j [6] but not available as an open-source tool. We found that among all the available graph-based technologies, Neo4j and DGraph [3] have rich set of algorithms and functionality to compare with standard RDBMS PostgreSQL. Also, Neo4j supports .ttl format for the RDF data, while we need to convert to the JSON format for the DGraph. To analyze two different types of data sources with rich set of provided graph algorithms, we have finalized our GDBMS choice to the Neo4j and DGraph.

The state-of-the-art survey for the graph tools [14] describes an extended survey taken for different graph softwares and technologies available for the research. It clearly states the issues faced by different graph DB users. The survey was carried out successfully by interviewing industrial experts and researchers who are using graph tools. The survey explains all the related features for different Graph DB tools, but in specific, analysis of query execution time for available technologies is not mentioned clearly. That motivates this work to perform the same. One of the application for genome analysis in life science presented in [17] works for analysis of different datasets to visualize the same in graph format. It helps to add indexing and caching features. The recent work [16] discusses available three types of algorithms in Neo4j and how it responds to queries. It explains all path based, centrality, and community detection algorithms for cipher-language.

The all the available RDF data management techniques have one common point of concern, which is query cost. This work is evaluated for the same purpose. The recent work [2] performs the same query processing approach for TPC-H benchmark and for five graph algorithms. However, this work is carried out mainly for query processing for sensor data, and experiments are evaluated based on query execution time and data loading time.

3 Query Processing for RDF Data

This section includes the details of framework for relational and graph-based language. The execution plan and format used in the implementation are also presented in this section.

3.1 Relational and Graph-Based Query Languages

Structured query language (SQL) is a standard RDBMS language. For RDF data, there will be three columns subject, predicate, and object. Self-join operation helps to run aggregate and join queries. The structure of the SQL query can be formed through three clauses: SELECT, FROM, and WHERE.

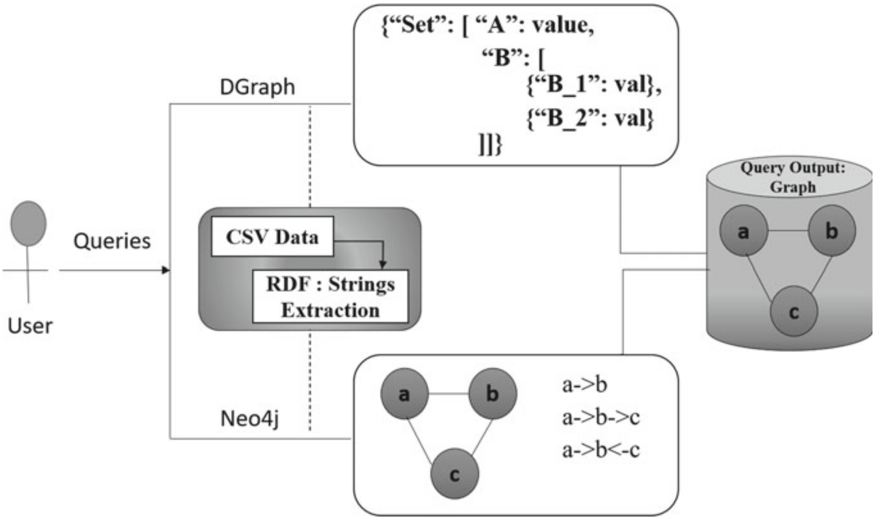


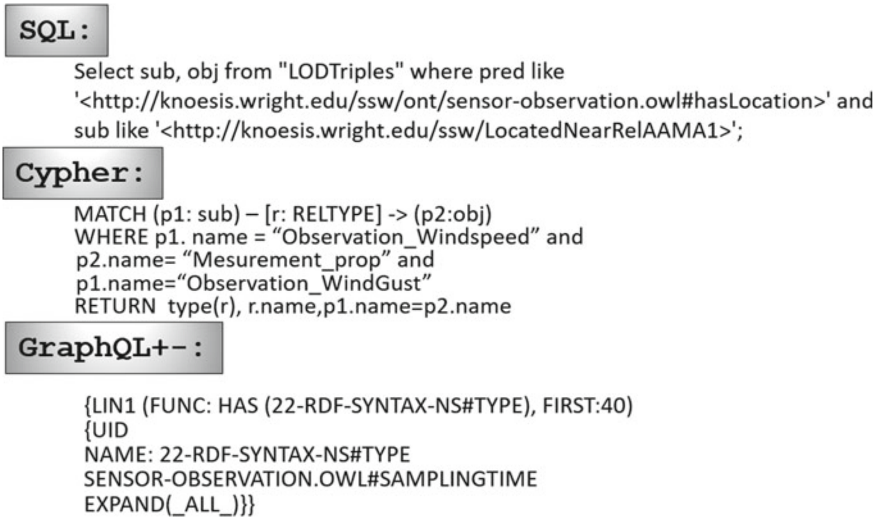
Fig. 1 Query processing in graph tools

In a GDBMS, vertices will be subjects and objects, and we need to form a relationships between them to traverse all the vertices for the given query. These relationships are formed through the edges between vertices of the graph. The query languages, GraphQL, cipher, SPARQL, are some of the examples to query with graph data. Cipher is a declarative graph query language based on the property graph model. Cipher is a simple but powerful language. Very complicated database queries can easily be expressed through cipher. Cipher-query defines in the pattern form of triplet as like RDF data. SELECT clause of SQL query language will be replaced by MATCH, and the connection will be formed by a relationship between two nodes.

GraphQL [4] is about asking for specific fields on objects. GraphQL would already be a very useful language for data fetching. GraphQL is written like a mutation block, and data are presented using JSON format. Edges which require to represent in graph need to be set in that block with faces on edges as a sub-block.

RDF data query processing using the graph-based techniques is described in Fig. 1. If a user registers a query to the system first, it will go under string extraction to make it compatible for the graph database. Then, it will be mapped with appropriate query clauses, and output will be displayed in the form of a graph. Figure 2 shows an example of a sensor data [5] query written in SQL, cipher, and GraphQL+−. As compare to SQL, graph queries define the relationship between subjects and objects in cipher whereas GraphQL+− has its own mutation block in which it works.

Postgres has been used as a RDBMS representative whereas Neo4j and DGraph are used as a GDBMS. DGraph uses GraphQL+− query language. The execution plan for both the relational and graph-based tools is depicted in Fig. 3. Postgres is a standard RDBMS which returns the result in the form of a table. DGraph has three main components: zero, alpha, and RatelUI. The zero works as a coordinator. Alpha



SQL:

```
Select sub, obj from "LODTriples" where pred like
'<http://knoesis.wright.edu/ssw/ont/sensor-observation.owl#hasLocation>' and
sub like '<http://knoesis.wright.edu/ssw/LocatedNearRelAAMA1>';
```

Cypher:

```
MATCH (p1: sub) – [r: RELTYPE] -> (p2:obj)
WHERE p1. name = “Observation_Windspeed” and
p2.name= “Mesurement_prop” and
p1.name=“Observation_WindGust”
RETURN  type(r), r.name,p1.name=p2.name
```

GraphQL+-:

```
{LIN1 (FUNC: HAS (22-RDF-SYNTAX-NS#TYPE), FIRST:40)
{UID
NAME: 22-RDF-SYNTAX-NS#TYPE
SENSOR-OBSERVATION.OWL#SAMPLINGTIME
EXPAND(_ALL_)}}
```

Fig. 2 Query examples

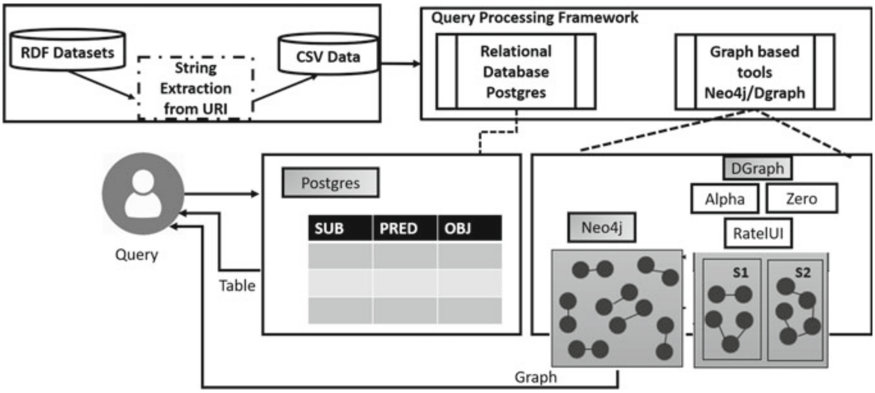
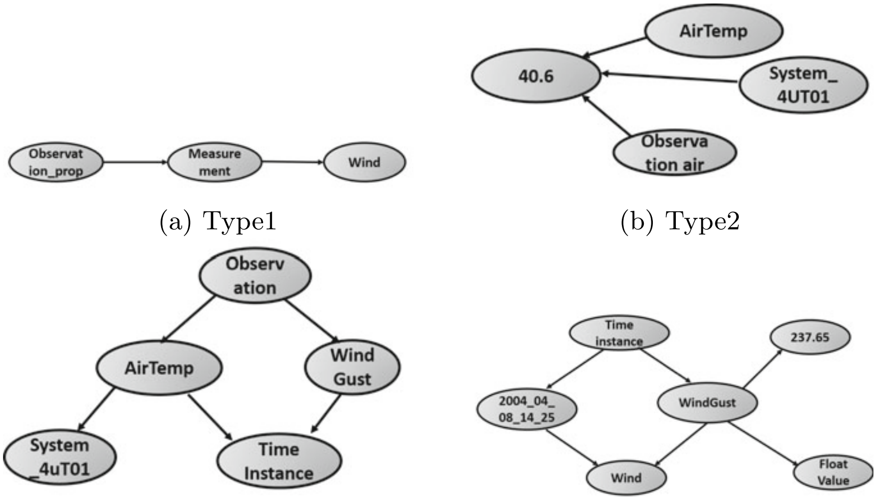

RDF Datasets
String Extraction from URI
CSV Data
Query Processing Framework
Relational Database Postgres
Graph based tools Neo4j/Dgraph
Query
Table
Postgres
SUB
PRED
OBJ
Neo4j
DGraph
Alpha
Zero
RatelUI
S1
S2
Graph

Fig. 3 Graph-based tools

is for database management, and RatelUI used for visualization. In one zero clusters, we have created three virtual nodes alpha_1, alpha_2, alpha_3. On a single host, these alphas communicate via different ports. Cipher-query execution is done using Neo4j.

For sensor data, there is information about weather is available, and the query will take place execution for each property of weather. For example, if the user wants to find out air temperature for a particular place at a specific time, then that query of both languages first will extract the information in the small string, and then, it will return the result in terms of a graph. For RDBMS, SQL will return the answer in the form of the table only. Cipher-query returns output in structure format, and



(a) Type1

(b) Type2

(c) Type3

(d) Type4

Fig. 4 Query types

it is possible to store the output in tabular as well as the graph format. DGraph is a distributed graph database which only returns the visualization of data. DGraph does not support tabular format. It supports JSON and RDF *N*-quad format of data.

4 Experimental Setup

This section includes details for the dataset and queryset used for the experiment. Hardware and software specifications used for the experiment have also been included within the same section.

4.1 Dataset and Query Set

LOD [5] (Linked Observation Data) is used as benchmark dataset for the experiment. Linked observation data have near about 3000 categories. It consists of weather observations from hurricanes in the US. The observations collected include measurements of phenomena such as temperature, precipitation, pressure, wind speed, and humidity in RDF format. There are 800k RDF triples used for this experiment from the available dataset.

For the query execution, we have used four types of queries. Type 1 queries are linear queries which select some predicates from data. Example for type 1 query is as follows: ***Find sensors relative humidity observed is "25.0"*** Type 2 queries are star queries which select specific subject/objects relevant to given predicate, like ***Find system name and location with their ids***. Type 3 queries are administrative or range queries which retrieve data using aggregation function or range function. Example for the same is as follows: ***Find average Wind speed at*** '<https://knoesis.wright.edu/ssw/LocatedNearRelA25>' ***in August 2004***. Type 4 queries are snowflake queries which are combination of both Type 1 and Type 2, like to ***Find location and id for a SystemA07***. Visualizations of all four types of benchmark LOD queries are shown in Fig. 4.

4.2 Hardware and Software

Implementation is done on system, Intel® Core (TM) i3-2100 CPU@3.10 GHz 3.10 GHz 8 GB. For Query execution Neo4j Desktop 1.1.10 is used and for visualization Neo4j [6] browser version 3.2.19 is used. We use the plugin NeoSemantics [7] to import the RDF data which is in turtle format and then convert it to graph-based architecture. To implement a distributed database on a single host, we use the DGraph [3] tool. Postgres, pgadmin4 have been used as a RDBMS.

5 Results

This section discusses results for all four types of queries and time taken by the relational and graph-based tools to load the available dataset.

5.1 Data Loading Time

We have started experiments by evaluating data loading time for both RDBMS and GDBMS. The results for the same are shown in Fig. 5. It has been observed that Neo4j requires more time to load the data than Postgres and DGraph. It also shows that DGraph and Postgres require almost similar amount of time. The reason is that DGraph has seperate components to handle data and query. Dgraph zero handles the data, and Ratel handles the queries. While Neo4j handles the data and query both as a centralized system.

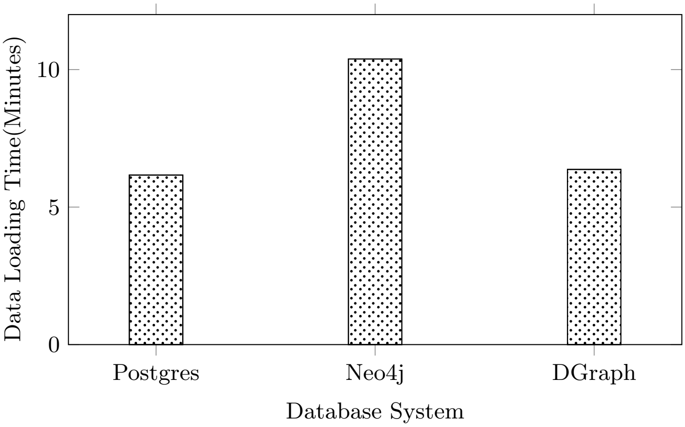


Fig. 5 Data loading time for RDBMS and GDBMS

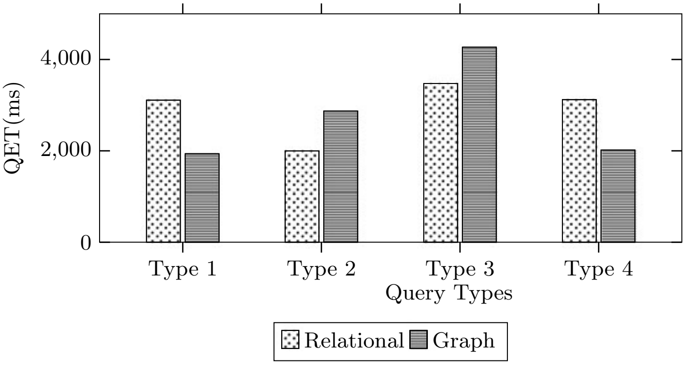


Fig. 6 Query execution time

5.2 *Query Execution Time (QET)*

For the evaluation of query execution time (QET), there are total 16 queries accelerated for both RDBMS and GDBMS. These 16 queries are divided among four types. The detailed information for the query types is available in the previous section. All the results for query execution are averaged over 3 consecutive hot runs. The average QET for respective query types has been shown in Fig. 7.

Type 1 queries are linear queries. It works to fetch the relevant subjects and objects from the data. For example, value for the temperature or windspeed for the particular location. As shown in Fig. 7, GDBMS outperforms RDBMS in terms of Type 1 queries.

To fetch the data, traversal of the graph is less costlier than SELECT operation of SQL. Type 2 queries are star queries. As per the results shown in Fig. 6, RDBMS

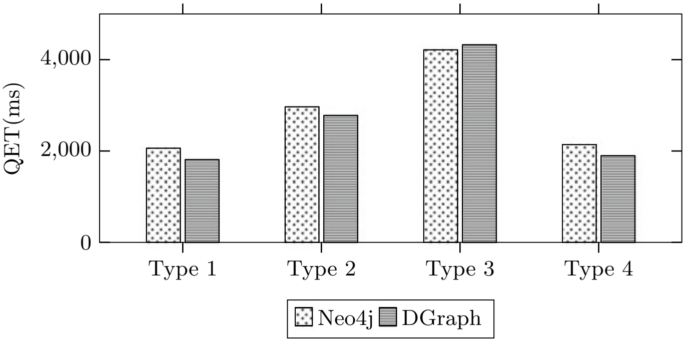


Fig. 7 Comparison Neo4j and Dgraph

outperforms for these types of queries. Star queries help to select subjects and objects based on predicate given in a query.

For the Type 3 queries, RDBMS works well than GDBMS. Aggregation operations are costly in GDBMS. Type 4 queries are snowflake queries. These types of queries suffer from costliest join operations. As shown in Fig. 6, GDBMS queries run faster than RDBMS queries. It can be observed that Type 2 and Type 3 queries perform better for RDBMS whereas GDBMS works well with Type 1 and Type 4 queries.

5.3 Comparison of Graph-Based Tools

As shown in Fig. 7, Neo4j works slower than DGraph in terms of data loading and QET. DGraph works 2.5% faster than Neo4j when averaged over all types of queries. The reason is that DGraph has seperate component, alpha to deal with data, and Ratel to handle queries whereas Neo4j handles everything on a central coordinator node. Neo4j shows result in terms of table also whereas DGraph only displays the graph visualization, and output will also remain as it is until we take snapshot. In Neo4j, we can store the result in PNG /CSV /SVG format.

5.4 Comparison RDBMS Versus GDBMS

As we have discussed in previous section, both relational and graph-based systems have their own advantages, and there is no boundary for the usage of the same. Figure 8 shows the average QET for all four types of queries. For the graph databases, we have averaged QET over both the tools. The result shows that on an average both approach reports the same query execution time.

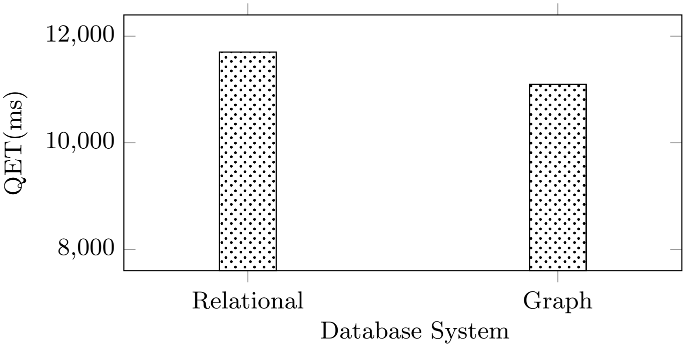
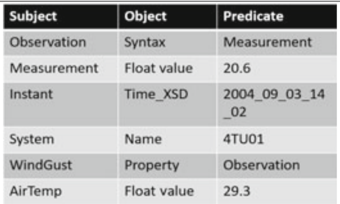
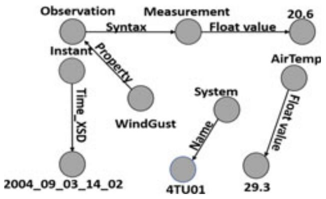


Fig. 8 Average query execution time for RDBMS and GDBMS

Table 1 Comparison of RDBMS and GDBMS

<table>
<tr><th>Parameters</th><th>RDBMS</th><th>GDBMS</th></tr>
<tr><td>Structure</td><td>Tables: rows and columns</td><td>Vertices and edges</td></tr>
<tr><td>Relationships</td><td>Represented by Foreign keys between tables</td><td>Represented by edges between vertices</td></tr>
<tr><td>QET</td><td>Efficient for star and range queries</td><td>Efficient for projection and join queries</td></tr>
<tr><td>Data loading time</td><td>Similar to Dgraph</td><td>Almost similar but Neo4j is taking longer than usual</td></tr>
<tr><td>Use cases</td><td>Mostly transaction and accountancy based Use cases</td><td>Fraud detection and recommendation engines</td></tr>
<tr><td>Example</td><td>
<table>
<tr><th>Subject</th><th>Object</th><th>Predicate</th></tr>
<tr><td>Observation</td><td>Syntax</td><td>Measurement</td></tr>
<tr><td>Measurement</td><td>Float value</td><td>20.6</td></tr>
<tr><td>Instant</td><td>Time_XSD</td><td>2004_09_03_14_02</td></tr>
<tr><td>System</td><td>Name</td><td>4TU01</td></tr>
<tr><td>WindGust</td><td>Property</td><td>Observation</td></tr>
<tr><td>AirTemp</td><td>Float value</td><td>29.3</td></tr>
</table></td><td>
</td></tr>
</table>

The comparison between relational and graph-based approach has been extend as shown in Table 1. The parameters for the comparison are stated in first column. The structure of RDBMS can be formed with a table, whereas for GDBMS, it can be visualized with edges and vertices formed through a graph. Mostly, RDBMS is easy to used for transactional and accountancy-related use cases, and GDBMS is popular to use with fraud detection and recommendation engine-related use cases. The example of mapping between these two for LOD [5] dataset is presented in the last row. Two state-of-the-art works DWAHP [9] and LBSD [10] are implemented for distributed RDF partitioning using relational and graph-based approach, respectively. It has been found that former requires 50% inter-node Communication (INC) in a distributed environment, whereas later reports 58% INC in a distributed environment.

6 Conclusion

To manage increasing size of RDF data, queryexecution is an important parameter to work upon. The analysis of RDF query processing is done based on experiment of SQL and NoSQL query language processing. This analysis has been carried out in detail using three tools specific for RDBMS and GDBMS. Postgres for RDBMS and for the GDBMS; Neo4j and DGraph. For the RDBMS, SQL has been used for the query processing. For the NoSQL query languages GraphQL+− and cipher. Neo4j supports cipher, and DGraph tool supports GraphQL+− query language. RDBMS outperforms GDBMS for the data loading. For the QET, the analysis result shows that DGraph is much faster than Neo4j as it does not require to display data tables and information. RDBMS and GDBMS both have their own advantages. This experiment concludes that complex join operation makes RDBMS slower while accelerating queries, whereas complex search makes graph database slower while traversing longest paths through all the vertices. So, for Type 1 and Type 4, GDBMS outperforms RDBMS, and for Type 2 and Type 3, RDBMS outperforms GDBMS. As a future work, we can work for the optimization of the cost of the queries.

References

1. Abadi DJ, Marcus A, Madden S, Hollenbach KJ (2007) Scalable semantic web data management using vertical partitioning. In: Proceedings of the 33rd international conference on very large data bases. University of Vienna, Austria, 23–27 Sept 2007. ACM, pp 411–422
2. Cheng Y, Ding P, Wang T, Lu W, Du X (2019) Which category is better: Benchmarking relational and graph database management systems. Data Sci. Eng. 4(4):309–322
3. DGraph set up. https://dgraph.io/
4. GraphQL tutorial. Available at https://www.graphql.com/tutorials/
5. Linked observation data. http://wiki.knoesis.org/index.php/LinkedSensorData
6. Neo4j desktop. https://neo4j.com/
7. Neosemantics. https://github.com/neo4j-labs/neosemantics/blob/4.2/README.md
8. Özsu MT (2016) A survey of rdf data management systems. Front Comput Sci 10(3):418–432
9. Padiya T, Bhise M (2017) DWAHP: workload aware hybrid partitioning and distribution of RDF data. In: Desai BC, Hong J, McClatchey R (eds) Proceedings of the 21st international database engineering and applications symposium, IDEAS 2017, Bristol, United Kingdom, 12–14 July 2017. ACM, pp 235–241
10. Pandat A, Gupta N, Bhise M (2021) Load balanced semantic aware distributed RDF graph. In: IDEAS 2021: 25th international database engineering and applications symposium, Montreal, QC, Canada, 14-16 July 2021. ACM, pp 127–133
11. Peng P, Zou L, Chen L, Zhao D (2019) Adaptive distributed rdf graph fragmentation and allocation based on query workload. IEEE Trans Knowl Data Eng 31(4):670–685
12. PostgreSQL setup. Available at https://www.postgresql.org/download/
13. RDF primer at https://www.w3.org/TR/rdf-primer/
14. Sahu S, Mhedhbi A, Salihoglu S, Lin J, Özsu MT (2020) The ubiquity of large graphs and surprising challenges of graph processing: extended survey. VLDB J 29(2–3):595–618
15. Shah B, Padiya T, Bhise M (2015) Query execution for rdf data using structure indexed vertical partitioning. In: 2015 IEEE international parallel and distributed processing symposium workshop, pp 575–584

16. Wiese L (2019) Data analytics with graph algorithms—a hands-on tutorial with neo4j. In: Meyer H, Ritter N, Thor A, Nicklas D, Heuer A, Klettke M (eds) Datenbanksysteme für business, Technologie und Web (BTW 2019), 18. Fachtagung des GI-Fachbereichs „Datenbanken und Informationssysteme" (DBIS), 4.-8. März 2019, Rostock, Germany, Workshopband. LNI, vol P-290. Gesellschaft für Informatik, Bonn, pp 259–261
17. Wiese L, Wangmo C, Steuernagel L, Schmitt AO, Gültas M (2018) Construction and visualization of dynamic biological networks: benchmarking the neo4j graph database. In: Auer S, Vidal M (eds) Data integration in the life sciences—13th international conference, DILS 2018, Hannover, Germany, 20–21 Nov 2018, proceedings. Lecture notes in computer science, vol 11371. Springer, pp 33–43
18. Zou L, Özsu MT (2017) Graph-based rdf data management. Data Sci Eng 2(1):56–70

Object-Detection Based Recommendation Engine for Advertising Using Deep Learning

Srinidhi Hiriyannaiah, Manish Manohar, Manas P. Shankar, D. S. Kaustubha, and Kaushik Kampli

Abstract With the exponential increase in online advertising, it has become increasingly important to determine new and innovative methods to identify the right advertisement to display to the consumer. Current methods of recommendations for advertisements employed by popular streaming platforms use the implicitly and explicitly collected data of the users for recommending advertisements. These recommendations may not always be accurate, and a user could be bogged down by a huge number of ads from irrelevant domains. Our research focuses on a novel approach for advertising which utilizes object detection for recommending advertisements. In its current state, this idea is based on the frequency of objects detected in the frames of the video. The main outcome was that our recommender engine performed better in terms of the relevancy of the advertisement, when compared to existing systems, most notable of which is YouTube. We also note that the privacy of the user is also improved, since their personal data is not being collected in order to recommend advertisements. In terms of future scope, we identify some key areas of improvement, such as the further classification of the objects detected into sub domains, making for more fine-tuned recommendations, as well as factors involving the selection of videos such as quality, duration, and relevance.

Keywords Object detection · Deep learning · Recommendation engine

1 Introduction

Object detection is capable of combining the tasks of classification of objects, as well as localization. Currently, detectors are split into two categories: Networks in charge of separating the tasks of determining the location of objects and then classifying them, and networks which predict bounding boxes and class scores at the same time. These networks find applications in fields ranging from real-time object detection

S. Hiriyannaiah · M. Manohar (✉) · M. P. Shankar · D. S. Kaustubha · K. Kampli
Department of Computer Science and Engineering, Ramaiah Institute of Technology, MSR Nagar, Bengaluru, Karnataka, India
e-mail: manish.m1138@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_42

in remote sensing images [1], to the detection of cultural heritage objects in high resolution airborne LIDAR data in Norway [3], to the detection of shuttlecocks for a badminton robot [4], as well as the detection of objects in rural roads [5], where the detection of road signs becomes a legitimate issue.

We see that recommender systems have evolved over time in two different directions: content-based filtering and collaborative filtering. Collaborative filtering is used to map (profile) the taste of users and offer content to them that users with similar preferences liked. This has been used to great effect in conventional video-based recommender systems used in popular streaming services [6]. In content-based filtering, we should know the dimensions of the entity to be recommended and the user's preferences for these dimensions. A prominent example for this is music recommendation, wherein recently the mood, and personality of the user has been shown to produce accurate and dynamic results [8].

Recommender systems that use multimedia data have come to find applications in fields such as online learning [10], OTT platforms, and social media platforms. The recommendation process follows a pipeline architecture. The first step is to break the video into frames for object detection. This is followed by the extraction of the important features from the frames and the representation of these features as a feature vector. Finally, based on what we have to recommend, we semantically orient the extracted features into target classes for recommendations [11].

However, these are not the only kinds of recommender systems that are used. There exists systems based on autoencoding of social representations [12], genetic algorithms [14], and numerous other context and content-based scenarios.

In this paper, we propose an object detection-based recommendation engine for advertising using deep learning. The central idea is that of a recommendation system which analyses the objects in a video and recommends the most relevant advertisement to the customer on the basis of the frequency of the objects detected in the frames of the video, as well as other parameters related to the context-defined business model.

The paper is organized as follows. Section 2 describes the recent developments in recommender systems using object detection and about the developments in the individual fields. Our proposed methodology is discussed in Sect. 3 followed by the results in Sect. 4. We conclude by stating the central idea of the paper and the main results in Sect. 5, followed by the references in Sect. 6.

2 Related Work

Most of the research into recommender systems has focused on finding the most accurate recommender system. However, other factors are also relevant, such as diversity, privacy, user demographics, robustness, as well as labelling. Object detection has also seen significant research be carried out in the fields of face detection and pedestrian detection, but it has significant utility in other areas of computer vision, including image retrieval and video surveillance. This section highlights some of the research endeavours into recommender systems and object detection.

Object detection has evolved to be able to detect small as well as similar objects. Reddy et al. [1] propose a method of using faster R-CNNs to handle real-time object detection in large volumes of remote sensing data. Noise filters and contrast enhancements were applied in the pre-processing step, and then subjected to training, which improved precision but reduced recall and accuracy due to false alarms. This was corrected by introducing a "misc" label for all false alarm objects, which in turn resulted in a precision of more than 90% on the C2S data and Resourcesat-2A satellite's AWiFS data. Dhillon et al. [2] provide a comprehensive review in the form of models, methodologies, and applications of convolutional neural networks to object detection. The working of CNN architectures is described in detail and are followed by detailed descriptions of models ranging from the classical LeNet model, to more recent models such as Xception, AlexNet, etc.

Trier et al. [3] demonstrate using R-CNNs in cultural heritage detection in local relief model (LRM) of airborne laser scanning (ALS) data. Charcoal kilns, grave mounds, and pitfall traps were semi-automatically detected in high-resolution LIDAR data. However, due to the relatively high false positive rate, it is desirable to limit the false positives found when this is applied to larger areas for detailed archaeological mapping, to reduce the amount of visual inspection necessary. Cao et al. [4] use novel variants of the prominent Tiny YOLOv2 detector, to detect shuttlecock positions for a badminton robot. Due to its fast movement in a complex context, accuracy, and computational efficiency emerge as two important parameters, especially in a real-time scenario. Modifications to the loss function and the architecture enable Tiny YOLOv2 to improve detection speed and retain more semantic information, leading to high detection accuracy with faster speed, compared to other prominent detectors like R-CNNs, SSD, Tiny YOLOv2, and YOLOv3.

Barba-Guaman et al. [5] make use of the YOLO and SSD models to carry out object detection in rural roads in different environments. Due to the difficulty in finding readily available data sets for such a problem, a new data set was created and was used to produce acceptable results, given important conditions such as lighting, viewing perspectives, and partial occlusion of the objects. Recommender systems make use of Collaborative Filtering (CF) as an intelligent search mechanism for personalized recommendations. Thakker et al. [6] carry out a comprehensive analysis of movie recommender systems that employ Collaborative Filtering, detailing approaches involving users as well as the items themselves. Crucial steps taken in the recent past have been outlined, as well as present challenges, such as cold starts, data sparsity, and scalability issues, along with the novel proposed methodologies to tackle them.

Zhang et al. [7] aim to provide a comprehensive review of research endeavours undertaken through the years in the field of deep learning-based recommender systems. The motivation to turn towards deep learning-based recommenders is due to the ability of deep learning to effectively capture nonlinear, as well as nontrivial relationships between users and items. It also captures the intricate relationships within the data itself, from abundant accessible data sources such as textual, contextual as well as visual information. Moscato et al. [8] tackle the improvement of music recommendations, based on Content-based Filtering. A technique based on personality traits, moods, and emotions identification of a single user, derived from analysing user behaviour in a social environment, is used. A profile and mood-based strategy has been shown to significantly improve the browsing of audio collections, compared to classical rating-based approaches.

Capelleveen et al. [9] discuss a challenging proposition, involving the design of recommender systems. Factors involved in this process include data quality, the prioritization of recommender system goals, such as precision, recall, accuracy, and the novelty of item suggestions. The overall aim is to build an ontological model for the design of more varieties of recommender systems, and this culminates in providing a novel perspective on recommender systems, centred around requirement specification, with the proposal of a canvas management tool to facilitate recommender system design. Self-Regulated learning [10] is the process in which the students have to learn the content being taught to them at their own will. In order to refine this process, recommender systems could be used to analyse the topics being learnt by the student currently and recommend similar videos. These recommendations have to be adaptive in order to cater to the needs of the student. With the exponential increase in the number of videos on various online platforms, it is becoming very important to have a mechanism to recommend videos to the users effectively.

A pipeline architecture is proposed in [11], which could be used for audio-based recommendations, image-based recommendations, and video-based recommendations. Pan et al. [12] propose an autoencoding model, capable of learning high-level social representations for the recommender system. The inability of current systems to model complex interactions between pairs of users on social media leads to a loss of information. An autoencoder model, named the Sparse Stacked Denoising Autoencoder (SSDAE) is developed to address this issue. Experiment results from the Epinions and Ciao data sets show that SSDAE can significantly improve the performance of recommender systems, particularly for sparse users.

Paradarami et al. [13] present a deep learning neural network framework that utilizes metadata information from reviews to generate predictions. A set that includes both content and collaborative-based features is shown to allow the development of a neural network model. This model minimizes log loss and rating misclassification errors using the stochastic gradient descent optimization algorithm. An interesting prospect is presented in the testing of a recommender system using these distributed processing frameworks. Alhijawi et al. [14] make use of a genetic algorithm to improve the CF-based recommender, utilizing semantic information, satisfaction levels, and historical rating data as the parameters. A methodology is introduced that evaluates lists of items using the above parameters, to find the best suited list for the active user, and recommend it. The results demonstrate the effectiveness of this methodology, and its ability to achieve more accurate predictions, irrespective of the number of K-neighbours.

There exists two main drawbacks of recommender systems, that of sparsity and scalability. The trade-off between accuracy and computation time in making recommendations needs to be made appropriately, since the systems need to produce recommendations that are accurate, while operating in real time. Nilashi et al. [15] hence develop a hybrid recommendation method based on Collaborative Filtering (CF) approaches, by making use of dimensionality reduction and ontology techniques. In order to find the most similar users and items in each cluster of users and items, Singular Value Decomposition (SVD) is used. This significantly improves the scalability of the recommendation system and the methods it uses. With the emergence of many online learning platforms, the need for solving the queries of students has increased. While watching a video if a student is stuck at some point, he could create a problem tag [16] and add a comment. The provider of the video could then look into this problem and provide additional learning materials that would speed up the learning process.

While dealing with movie recommendations, it would be a better idea to use a fusion of multiple user representations such as the latent, item-level, category-level, and neighbour-assisted representations [17] in order to deal with the volatility of user's interests. Combining these representations prior to recommendation provides better quality recommendations. Content-based recommendations could be used for converting a given video recommendation problem into a similarity problem [18]. In order to achieve this, the authors have used a five layer deep neural network to create the embeddings of the videos based on the visual and audio features.

Tohidi et al. [19] propose a hybrid approach using clustering and evolutionary algorithms to improvise the recommendations. A cyclical method of clustering and classification is used to recommend the videos to the user based on their cluster. Reinforcement learning could be used for recommending advertisements, as proposed by Zhao et al. [20], wherein a novel Deep Q-network architecture is developed for recommending ads. The reward function does not only focus on ad revenue but also the user experience and relevance as well. It is capable of balancing the frequency as well as the appropriateness of the ads.

3 Proposed Methodology

In this section, we describe the proposed methodology of the recommender engine and the overall system. Figure 1 depicts the overall design of the system, detailing the flow of data between different components.

3.1 Data Collection

Due to the lack of a dedicated data set of detected objects in ordinary videos found on popular social media platforms, we were motivated to collect a varying set of videos

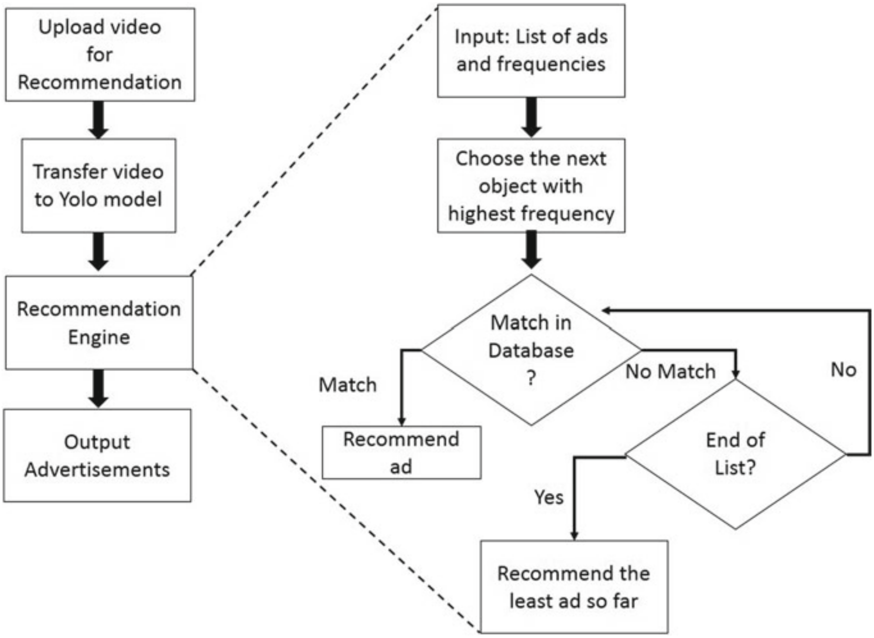

Upload video for Recommendation
Transfer video to Yolo model
Recommendation Engine
Output Advertisements
Input: List of ads and frequencies
Choose the next object with highest frequency
Match in Database ?
Match
No Match
Recommend ad
End of List?
No
Yes
Recommend the least ad so far

Fig. 1 Proposed system for object detection-based recommendation

to perform object detection. Google Colab, a free cloud service providing access to GPUs, was used to train the object detection model, to detect the objects belonging to various domains.

A sample of 40 videos were obtained from different domains and subdomains. The videos were sourced primarily from YouTube, since it happens to be the largest video sharing and second largest search engine in the world. The videos themselves feature objects in a day-to-day scenario, with our goal being to test the validity and viability of the recommender system in recommending advertisements relevant to what occurs in the natural environment of the video being viewed.

3.2 Object Detection

The YOLOv3 model is used for object detection. The YOLO algorithm needs only one forward propagation pass through the convolutional neural network (CNN) to make predictions which is why it "only looks once" at the input image. The algorithm applies a single CNN to the whole frame of the video and repeats this for every consecutive frame. The image contained in that frame is divided into a grid of 13 by 13 cells where each of these cells is responsible for predicting five bounding boxes. This in turn generates the bounding boxes and also predicts the probabilities for each region. These bounding boxes are then weighted by the predicted probabilities. To calculate the classification loss for each label, the model uses binary cross-entropy. Logistic regression predicts the class predictions and object confidence.

The architecture used as the backbone of the YOLOv3 model is called DarkNet-53 as indicated in Fig. 2. Its primary function is to perform feature extraction. It has 53 layers of convolutions. For each operation, we have a sequence of layers starting with a convolution layer, which is used to convolve multiple filters on the images and produce multiple feature maps, followed by Batch Normalization layers and a leaky ReLU activation. A convolutional layer with stride 2 is used to down sample the feature maps, which helps in preventing loss of low-level features often attributed to pooling. The inputs are 448 × 448 images.

3.3 Recommendation

In this section, we discuss the algorithms used for the recommendations using the objects detected in the videos. In algorithm 1, the first part of the input consists of the list of objects and their frequencies retrieved from the database, where they were stored after extracting them from the YOLO model. The second part of the input consists of the list of ads provided by the companies. This comes with attributes such as the company description, the product description, amount paid for advertising, all of which are criteria in recommending ads.

To begin with, we traverse the dictionary of detected objects from highest to lowest frequency. We select the object which has both the highest possible frequency and which is present in the list of ads. This criteria may be extended if there are multiple ads for the same product, to include the amount paid by each company, etc. The case where none of the detected objects match the list of available ads, is addressed by recommending the ad that has been recommended the least number of times. We track this by maintaining an attribute that increments whenever the ad is recommended. Finally, the selected ad is sent back to the website, and the "times_recommended" attribute is incremented to reflect the same.

Algorithm 1: Recommender Engine
INPUT: List of objects [ob] containing frequency (no. of frames of occurrence in video), list of ads [adv] given by the ads agency
OUTPUT: Advertisement video chosen by RECOMMENDER ENGINE

(continued)

(continued)

```
BEGIN RECOMMENDER ENGINE
: Traverse highest frequency to lowest frequency object in [ob]
switch (object)
case MATCH: if in [adv], recommend ad
else continue
case NO_MATCH: No match found in [adv],
recommend ad that is recommended least times
: Update count of ads shown, in [adv]
END RECOMMENDER ENGINE
```

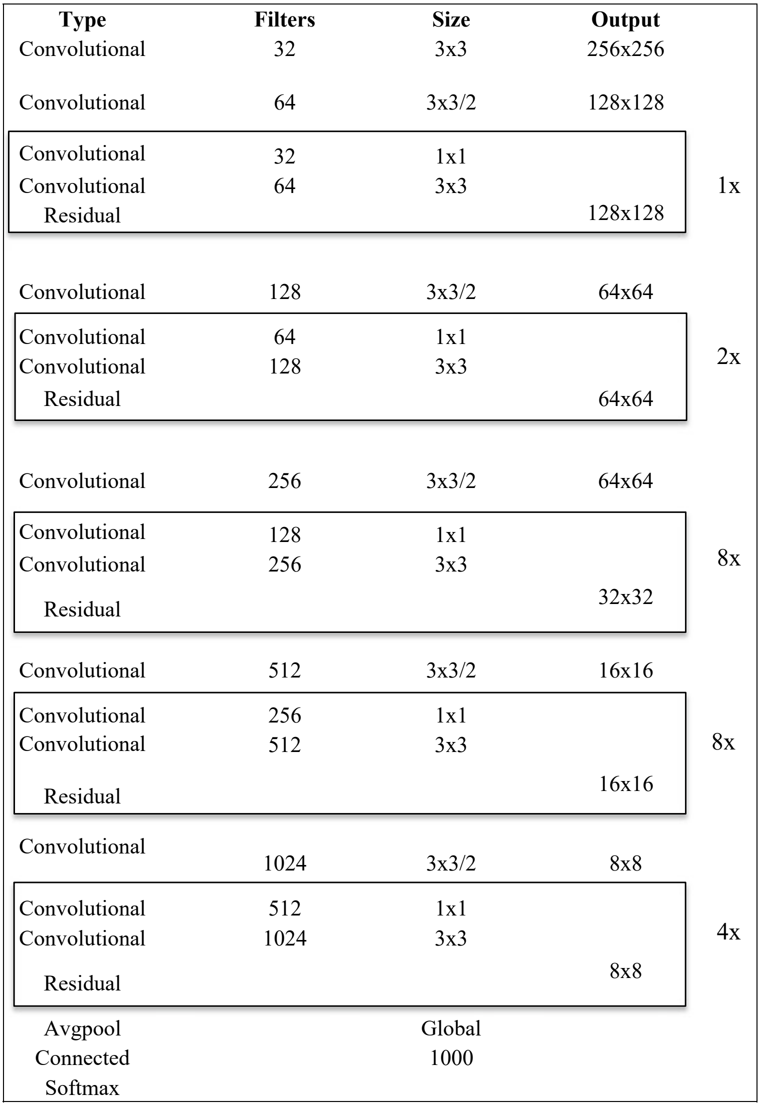
```
Algorithm 2: Ad Recommender
INPUT: video x uploaded to website, list of ads [adv] provided by advertisers
OUTPUT: Advertisement video y relevant to uploaded video
BEGIN AD RECOMMENDER
: Transfer x to YOLO model
: Execute YOLO model on x
: Extract labels l and count_of_labels c in each frame of x
: Store l, c to Database[primary key -> l, c]
: Execute RECOMMENDER ENGINE, inputs [l, c, [adv]]
: Return y from RECOMMENDER ENGINE
END AD RECOMMENDER
```

In algorithm 2, the video uploaded to the website by the user is taken as the input. The final output is the advertisement that is most relevant to the uploaded video. We begin by transferring the uploaded video to the YOLO detection model via appropriate HTTPS protocols. The model is then run on the video and labels of objects are extracted that indicate the count of each detected object in each frame of the video. These labels and their counts are then stored in the database to be processed by the recommender engine.

Now we move to the scenario where a video is played on the website. Here, we retrieve the dictionary containing the labels and the corresponding frequency for the video from the database and send it to the recommender engine, which selects the most appropriate advertisement on the basis of algorithm 1. Finally, the chosen advertisement is then displayed to the user after the played video ends in the website. This concludes the recommendation algorithm.

4 Experiments and Results

In this section, we describe the parameters of the tests carried out during object detection and recommendation. We outline statistics of the video samples, the recommended domains, and the mismatch in the recommendations made, as well as attempt to understand the reasons for these mismatches.



Type	Filters	Size	Output	
Convolutional	32	3x3	256x256	
Convolutional	64	3x3/2	128x128	
Convolutional	32	1x1		1x
Convolutional	64	3x3		
Residual			128x128	
Convolutional	128	3x3/2	64x64	
Convolutional	64	1x1		2x
Convolutional	128	3x3		
Residual			64x64	
Convolutional	256	3x3/2	64x64	
Convolutional	128	1x1		8x
Convolutional	256	3x3		
Residual			32x32	
Convolutional	512	3x3/2	16x16	
Convolutional	256	1x1		8x
Convolutional	512	3x3		
Residual			16x16	
Convolutional	1024	3x3/2	8x8	
Convolutional	512	1x1		4x
Convolutional	1024	3x3		
Residual			8x8	
Avgpool		Global		
Connected		1000		
Softmax				

Fig. 2 Darknet-53 architecture [22]

4.1 Sampling and Object Detection Results

A sample of 40 videos were considered from varying overlapping domains. While some videos prominently featured one or two domains, it was considered infeasible to ignore the presence of objects from other domains.

Table 1 Object detection for different domains

Domain	No. of samples containing content relevant to this domain	Total no. of objects detected in these samples	No. of detected objects of this particular domain
Consumer electronics	22	533	145
Lifestyle	19	437	208
Food	5	123	29
Kitchenware	6	124	37
Sports	4	60	16
Automobiles	7	151	35

For this reason, a count of the number of videos containing a considerable number of objects of each particular domain was carried out. Several videos overlap with regard to the objects they contain and the domains these objects belong to. In order to better understand the presence of objects of a particular domain, the total number of objects detected in each video, as well as the number of objects detected belonging to a particular domain was determined. These statistics have been summarized in Table 1.

Figure 3 contains snapshots of bounding boxes generated by the model on a frame of the video being processed, giving a glimpse into the objects being detected in every frame. The figure contains instances of the model at work on objects belonging to different domains.

4.2 Recommendation Results

We now analyse the recommendations made by our system and compare it with existing recommendations of advertisements made by various online video streaming platforms. In Table 2, a comparative study of the recommended advertisements by our system and that of YouTube has been presented. To ensure fairness of results and maintain a sense of objectivity, a survey was conducted on numerous people regarding the nature of advertisements being played after a particular video sample was viewed by them on YouTube. A survey conducted as part of a research endeavour into communication effectiveness of YouTube advertising revealed a significant positive correlation between informativeness of the advertisement and brand recognition of the advertiser [21]. What this implies is that users are more likely to watch an advertisement if they find it to be more informative in terms of the product they are interested in. This happens to be the foundation for the evaluation of our recommender system.

The third column of Table 2 summarizes the results of Table 1. It depicts the fraction of objects detected in the video samples that actually belonged to the domain under consideration. On average, it was determined that in samples containing content

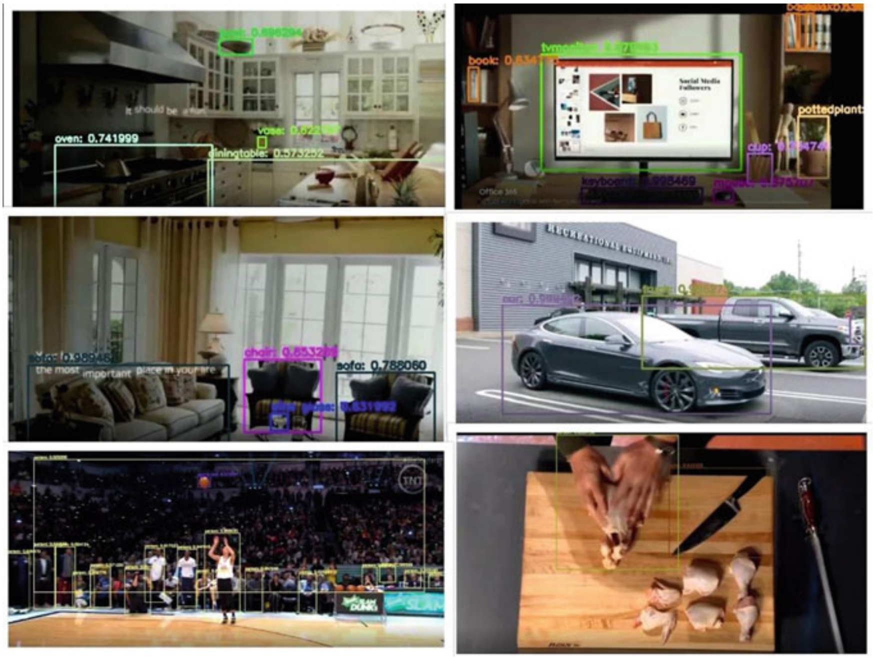


Fig. 3 Objects detection in sample videos

Table 2 Recommendations with object detection

Domain	No. of samples containing content relevant to this domain	Fraction of detected objects in the samples belonging to this domain	No. of recommendations made by the proposed system matching this domain	No. of recommendations made by existing systems [21] matching this domain
Consumer electronics	22	0.27	15	7
Lifestyle	19	0.48	10	4
Food	5	0.24	2	4
Kitchenware	6	0.29	4	2
Sports	4	0.27	2	2
Automobiles	7	0.24	5	3

relevant to a particular domain, 29.8% of detected objects in those samples actually belonged to that domain.

The third and fourth columns summarise the recommendations made by our system and that of YouTube's AdSense. Depicted in these columns is the number of recommended ad domains that were a match to the particular domain of the video under consideration.

The mismatches in some of the recommendations with respect to the domain under consideration were analysed. The root cause was found to be the occurrence of items belonging to other domains in other frames of the video. Since the recommender system in its current state uses a limited number of parameters for recommending ads (mainly the frequency of occurrence of objects), we find that mismatches do occur.

However, Table 2 tells us that the recommended ad domains by our recommender system are indeed more relevant to the video being played at the moment when compared to that of YouTube.

The number of mismatches could be reduced by training the object detection model to detect the domain to which a particular object belongs to. Currently, the Coco data set does not contain data related to the domain of the objects.

4.3 Performance Analysis

In order to evaluate the performance of our system, we calculated a relevance score, taking the fraction of videos for which an advertisement matching the domain was recommended. This data was taken from Table 2. The equation to calculate the relevance score, specifying the use of the data from Table 2 is as shown in Eq. 1.

$$\text{Relevance Score} = \frac{\text{Recommendations matching a domain}}{\text{No of samples relevant to domain}} * 5 \tag{1}$$

As is evident in Table 3, out of the six domains that were considered, our recommender system has better relevancy in five out of the six considered domains.

Table 3 Relevance scores of recommendations

Domain	No. of samples containing content relevant to this domain	Relevance score of recommendations made by proposed system	Relevance score of recommendations made by YouTube [21]
Consumer electronics	22	3.41	1.59
Lifestyle	19	2.63	1.05
Food	5	2.00	4.00
Kitchenware	6	3.33	1.67
Sports	4	2.57	2.50
Automobiles	7	3.57	2.14

5 Conclusion

In this paper, we have proposed a novel method for advertising, by making use of object detection for recommending advertisements to the users. Apart from the higher levels of relevancy of the advertisements displayed to the user, with respect to the video being played, privacy is improved since there is next to no form of data collection of the user. The object detection model and the parameters used for the recommendation system were mainly reliant on the frequency of detected objects. It was found from our experiments and observations that the recommender system is indeed able to make relevant recommendations. For future research, the identification of domains and possibly subdomains, for the detected objects will provide additional parameters for the recommender system, making it more context-aware, and hence able to reduce the mismatches in the recommended domains and subdomains. Along with the consideration of domains and subdomains, other important business parameters would also need to be considered in a production environment, in terms of which customers and which products to prioritize for a given detected object in a particular domain, even taking into account factors like the region where the system is being used, the click-through rate of a recommended ad, and so on. Therefore, such factors would provide concrete information regarding the optimal environment where the recommender system could be used to its greatest effect.

References

1. Reddy VB, Pramod Kumar K, Venkataraman S, Raghu Venkataraman V (2020) Real-time object detection in remote sensing images using deep learning. In: Hassanien A, Bhatnagar R, Darwish A (eds) Advanced machine learning technologies and applications. AMLTA. Advances in intelligent systems and computing, vol 1141. Springer, Singapore
2. Dhillon A, Verma GK (2019) Convolutional neural network: a review of models, methodologies and applications to object detection. Progr Artif Intell
3. Trier ØD, Reksten JH, Løseth K (2021) Automated mapping of cultural heritage in Norway from airborne lidar data using faster R-CNN. Int J Appl Earth Observ Geoinform 95:102241. ISSN 0303-2434
4. Cao Z, Liao T, Song W, Chen Z, Li C (2021) Detecting the shuttlecock for a badminton robot: a YOLO based approach. Expert Syst Appl 164:113833. ISSN 0957-4174
5. Barba-Guaman L, Naranjo JE, Ortiz A, Gonzalez JGP (2021) Object detection in rural roads through SSD and YOLO framework. Adv Intell Syst Comput 1365 AIST:176–185
6. Thakker U, Patel R, Shah M (2021) A comprehensive analysis on movie recommendation system employing collaborative filtering. Multimed Tools Appl 80:28647–28672
7. Zhang S, Yao L, Sun A, Tay Y (2019) Deep learning based recommender system. ACM Comput Surv 52(1):1–38
8. Moscato V, Picariello A, Sperli G (2020) An emotional recommender system for music. IEEE Intell Syst
9. Van Capelleveen G, Amrit C, Murat Yazan D, Zijm H (2020) The recommender canvas: a model for developing and documenting recommender system design. Expert Syst Appl
10. Du J, Hew KFT (2021) Using recommender systems to promote self-regulated learning in online education settings: current knowledge gaps and suggestions for future research. J Res Technol Educ 1–22

11. Deldjoo Y, Schedl M, Cremonesi P, Pasi G (2020) Recommender systems leveraging multimedia content. ACM Comput Surv (CSUR) 53(5):1–38
12. Pan Y, He F, Yu H (2020) Learning social representations with deep autoencoder for recommender system. World Wide Web 23:2259–2279
13. Paradarami TK, Bastian ND, Wightman JL (2017) A hybrid recommender system using artificial neural networks. Expert Syst Appl 83:300–313
14. Alhijawi B, Kilani Y (2020) A collaborative filtering recommender system using genetic algorithm. Inf Process Manage 57(6):102310. ISSN 0306-4573
15. Nilashi M, Ibrahim O, Bagherifard K (2018) A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques. Expert Syst Appl 92:507–520
16. Lehmann A (2019) Problem tagging and solution-based video recommendations in learning video environments. In: IEEE global engineering education conference (EDUCON), IEEE, pp 365–373
17. Chen X, Liu D, Xiong Z, Zha ZJ (2020) Learning and fusing multiple user interest representations for micro-video and movie recommendations. IEEE Trans Multimedia 23:484–496
18. Ferreira F, Souza DR, Moura I, Barbieri M, CV Lopes H (2020) Investigating multimodal features for video recommendations at globoplay. In: Fourteenth ACM conference on recommender systems, pp 571–572
19. Tohidi N, Dadkhah C (2020) Improving the performance of video collaborative filtering recommender systems using optimization algorithms. Int J Nonlinear Anal Appl 11(1):483–495
20. Zhao X, Gu C, Zhang H, Yang X, Liu X, Liu H, Tang J (2021) DEAR: deep reinforcement learning for online advertising impression in recommender systems. In: Proceedings of the AAAI conference on artificial intelligence, vol 35, no 1, pp 750–758
21. Anthony SJ, Liu V, Cheng C, Fan F (2020) Evaluating communication effectiveness of youtube advertisements. Int J Inform Res Rev 7(4):6896–6901
22. Farhadi A, Redmon J: Yolov3: an incremental improvement. In: Computer vision and pattern recognition, pp 1804–02767

Machine Learning-Based Scene Classification Using Thepade's SBTC, LBP, and GLCM

Sudeep D. Thepade and Mrunal E. Idhate

Abstract Photographs have become a necessary part of human life. Images are being captured all across the world. One of the significant issues in scene classification is that the photographs are taken in a variety of locations. There are many diverse scenes in these locations, such as an airport, a restaurant, a cafeteria, and a forest. Indoor scene classification and outdoor scene classification are the most common types of scene categorization. Scene classification is challenging, which poses issues in research and image processing. Image classification becomes challenging because of the numerous distinct items present in various sceneries, causing a machine to become confused about the scene's categories. The current challenge with scene classification results in low-classification accuracy. The object comes in various shapes and sizes and can be found in various locations, which is one of the reasons for the scene classification's poor accuracy. We will perform feature extraction for pattern recognition and multiple machine learning models for image categorization using the intel image classification dataset. Thepade's SBTC *n*-ary feature extraction approach for RGB and LUV color plane models and fusion LBP and GLCM is presented in this study. When the three individual feature extraction methods are compared with the fusion of Thepade's SBTC and GLCM with RGB color plane, the fusion of Thepade's SBTC and GLCM with RGB color plane yielded greater accuracy.

Keywords Scene classification · Feature extraction · Thepade's SBTC · Local binary pattern · GLCM

S. D. Thepade · M. E. Idhate (✉)
Department of Computer Engineering, Pimpri Chinchwad College of Engineering,
Pune 411044, India
e-mail: mrunalidhate@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_43

1 Introduction

One of the critical issues in scene classification. The photographs are taken in a variety of places. There are many diverse scenes in these locations, such as buildings, cafes, restaurants, and mountains. Indoor scene classification and outdoor scene classification are the most common types of scene categorization.

One of the most imperative topics in the world of image processing is scene classification. Image classification becomes challenging because of the numerous distinct items present in various sceneries, causing a machine to become confused about the scene's categories. The current challenge with scene classification results in low-classification accuracy. The object comes in a variety of shapes and sizes can be found in various locations, which is one of the reasons for the indoor scene classification's poor accuracy.

Several existing approaches such as deep learning, sparse representation, CNN, global and local feature extraction, and machine learning (ML) techniques are used for scene categorization. Thepade's Sorted Block Truncation Coding (Thepade's SBTC) is used in the multiple color plane models of the dataset in this research. The goal of the suggested system is to

- For feature extraction, use the N-ary Thepade SBTC algorithm.
- The Thepade's SBTC is applied to the RGB and LUV color planes of Kekre [1].
- For feature extraction, Thepade's SBTC is utilized in conjunction with local binary pattern (LBP), and other machine learning techniques are used for classification.

These sections are there in the paper. The introduction and need for the proposed system are covered in Sect. 1. Section 2 stands for the literature review. The proposed system is described in Sect. 3. The experimental environment and dataset are referred to as Sect. 4. Section 5 stands for outcome and discussion, while the final portion is for a conclusion.

2 Literature Survey

Several methodologies are used to propose the scene classification of the images. The subject of scene classification is broad. Indoor scene classification and outdoor scene classification are the most common types of scene categorization. The categorization of outdoor scenes is the topic here. The few currently accessible techniques for scene classification are explored in this section.

The scene classification techniques based on the support vector technique were presented by Jashandeep Singh Gill and Amanpreet Singh Brar. SIFT, SURF, and Tamura-based feature extraction algorithms are used in this research. The collected features are classified using the support vector machine. The MIT-67 dataset is used in this experiment [2].

Khan Salman and colleagues suggested an indoor scene classification approach based on mid-level feature extraction with a support vector machine and a convolutional neural network classification. Several datasets are used to test the approach. The MIT-67 dataset has a 71.2% accuracy rate [3].

Scene-recognition approaches based on sparse representation had been proposed by N. Sun and others. The RCNN training set is used as an existing external training set in the supplied system to detect the image's category using low-level feature extraction. Sparse recognition and sparse dictionary building are used to create a classifier to decide the class of the final output image after the mid-level feature extraction is completed [4].

A scene classification method based on the concatenation of global and local features was proposed by Douik A. et al. The image's global and local features are extracted using the RGB color channel. Several datasets are used to test the approach. The MIT-67 dataset has an accuracy of 82.35% [5].

Given the image classification method employing DCT and TSBTC methodologies, S. D. Thepade and et al. This paper determines the *n*-ary using the DCT on the augmented images. The TSBTC 4-ary [6] yields the best results. Another way for splicing identification is to use TSBTC with LBP with the help of different ML classifiers [7]. The approach for TSTBTC's similarity measure for content-based video retrieval [8], in which the Sorensen distance similarity measure outperforms the others [9].

Multileveled block truncation coding was employed by H. B. Kekre and et al. to improve content-based image retrieval, and the results were better than other current systems [10]. The same author also proposes a method for colorizing grayscale photos using the birthogonal color space with Kekr's rapid codebook generators [9].

S. R. Badre and colleagues proposed TSBTC to extract keyframes from video frames using the Haar wavelet. Wavehedge distance, Sorensen distance, mean square error, Alias Canberra distance, and Euclidean distance similarity measurements are used to evaluate the performance of the supplied approach [11].

3 Proposed Method

In this research, three approaches for feature extraction are proposed, all of which are based on Thepade's SBTC, GLCM, and LBP methodologies. These techniques are used on images in the RGB color plane and Kekre's LUV color plane. Upon feature extraction, various machine learning algorithms are used to classify an image into its appropriate category.

The first method is Thepade's SBTC; this method calculates the feature based on the pixel's intensity. There are two types of color planes. In this method, after the color plane matrix is extracted from images, each color matrix is converted into one dimensional (1D) array for further process. The ascending sorting method was applied on each plane's 1D array, and further, *N*-ary block truncation was used for pattern extraction (Fig. 1).

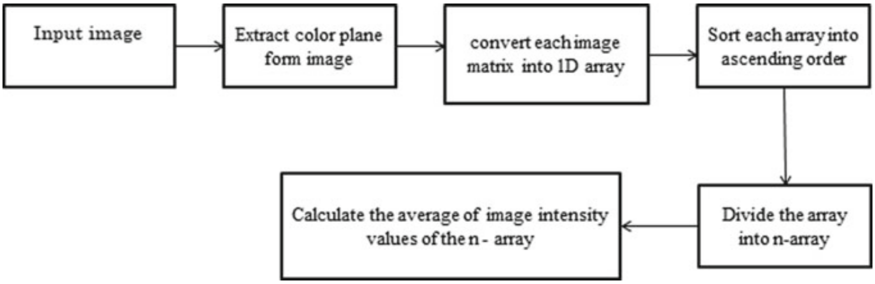
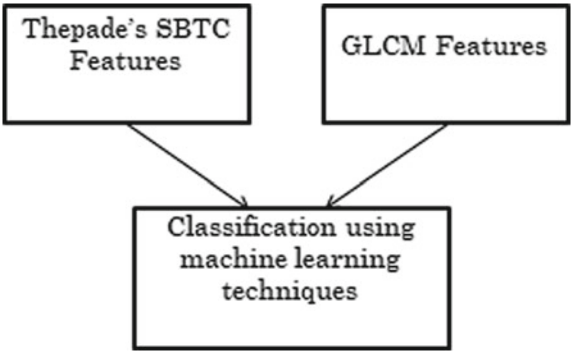

Input image
Extract color plane form image
convert each image matrix into 1D array
Sort each array into ascending order
Divide the array into n-array
Calculate the average of image intensity values of the n - array

Fig. 1 Flow diagram of Thrpade's SBTC techniques

Fig. 2 Block diagram of proposed fusion of Thespade's SBTC and GLCM



The second method is a fusion of Thepade's SBTC with GLCM on the RGB color plane. In this method, after the color plane matrix is extracted from images, each color matrix is converted into one dimensional (1D) array for further process. On that, each plane's 1D array ascending sorting method was applied, and further, *N*-ary block truncation was used for pattern extraction. Also, find the GLCM features for the same image set and merge both GLCM and Thepade's SBTC features for classification (Figs. 2 and 3).

The third method is a fusion of Thepade's SBTC and LBP with GLCM on the RGB color plane. In this method, before extraction, the color plane matrix from images LBP is applied on the input image, each color matrix converted into one dimensional (1D) array for further process. On that, each plane's 1D array ascending sorting method was applied, and further, *N*-ary block truncation was used for pattern extraction. Also, find the GLCM features for the same image set (Fig. 4).

Following the extraction of features, multiple machine learning classification algorithms are used, and the classification accuracy for the two approaches is calculated. The first method is to classify data using ten cross validations, while the second method is to divide data into 70–30 training and testing data.

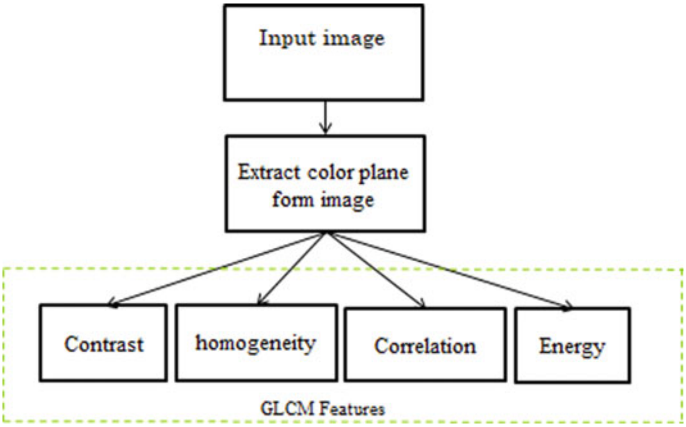
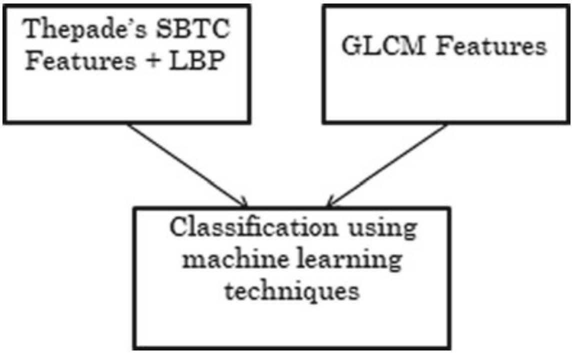


Fig. 3 Flow diagram of GLCM feature extraction

Fig. 4 Block diagram of proposed fusion of Thespade's SBTC and LBP with GLCM


Thepade's SBTC Features + LBP
GLCM Features
Classification using machine learning techniques

4 Experimental Environment

The approach is applied to the intel image classification dataset, which has six categories and 15232 photos. It features images of the building, street, sea, glacier, forest, and mountain scenes. The number of photographs in each category varies between 2000 and 2500. We used 100 photos per class, divided into six categories, for this investigation. Image classification dataset photos are shown in Fig. 5.

$$\text{Confusion Matrix} = \begin{bmatrix} \text{TN} & \text{FP} \\ \text{FN} & \text{TP} \end{bmatrix} \quad (1)$$

Where, TN = True negative result TP = True positive result FP = False positive result FN = False negative result

Accuracy = (TP + TN)/(TP + TN + FP + FN) * 100

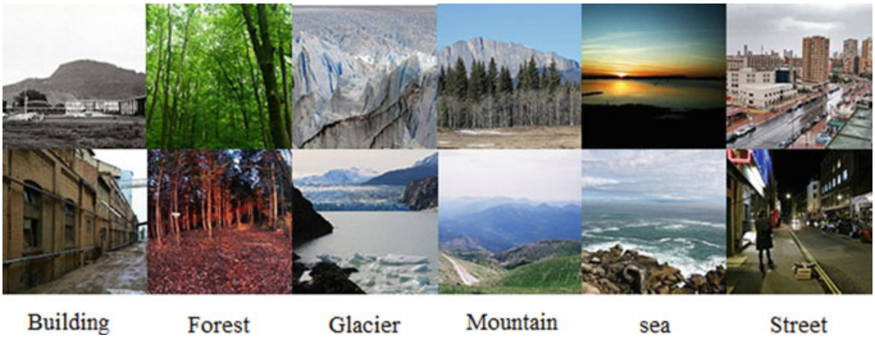


Fig. 5 Images from intel image classification dataset [12]

To assess performance, the corresponding values for accuracy are determined. A confusion matrix is required to calculate these values. MATLAB and WEKA are used to calculate the final figures.

5 Result and Discussion

Thepade's SBTC, LBP, and GLCM methods were used on the intel image classification dataset. The Thepade's SBTC is applied to the RGB and LUV color planes, respectively. The image's average intensity values of the separated RBG plane and LUV plane are calculated using N-ary Thepade's SBTC in this paper for 4-ary, 7-ary, and 10-ary.

Various machine learning methods such as random tree, random forest, simple logistic algorithm, logistic algorithm, Naive Bayes, Bayes net, and multilayer perception were applied after calculating average intensity values with Thepade's SBTC n-ary for 4-ary, 7-ary, and 10-ary. These machines are used to calculate the learning algorithm's accuracy and to evaluate the experiment's outcome. Data from 10 fold cross validation and 70–30 training–testing are used to calculate classification accuracy.

The Thepade's SBTC is applied on the RGB color plane in Tables 1 and 2. Thepade's SBTC is compared to the proposed approach is given in Tables 1 and 2. Thepade's SBTC has the maximum accuracy of 73% in 4-ary using a logistic classifier and multilayer perceptron while merging Thepade's SBTC with GLCM yields a greater accuracy of 83% in 7-ary using the same classifier 10-fold cross validation.

Whereas the highest accuracy of Thepade's SBTC for 70–30 training–testing data is 74% in 4-ary using a logistic classifier, fusing Thepade's SBTC with GLCM gives better accuracy of 82% for logistic classifier and 84% in multilayer perceptron classifier is 7-ary.

Table 1 Thepade's SBTC and Thepade's SBTC with GLCM applied on RGB color images for 10 fold cross validation in percentage

Algorithms	Thepade's SBTC 4-ary	Thepade's SBTC 4-ary + GLCM	Thepade's SBTC 7-ary	Thepade's SBTC 4-ary	Thepade's SBTC 10-ary	Thepade's SBTC 10-ary + GLCM
Random tree	61	69	51	63	45	56
Random forest	69	80	63	79	49	67
Simple logistic	72	82	66	81	58	72
Logistic	73	81	67	83	57	71
Naive Bayes	64	75	61	75	52	66
Bayes net	63	78	60	77	44	69
Multilayer perceptron	73	78	69	83	57	68

Table 2 Thepade's SBTC and Thepade's SBTC with GLCM applied on RGB color images for 70–30 training data in percentage

Algorithms	Thepade's SBTC 4-ary	Thepade's SBTC 4-ary + GLCM	Thepade's SBTC 7-ary	Thepade's SBTC 4-ary	Thepade's SBTC 10-ary	Thepade's SBTC 10-ary + GLCM
Random tree	65	64	59	58	48	64
Random forest	63	72	58	78	51	74
Simple logistic	71	81	65	51	59	75
Logistic	47	80	63	82	59	77
Naive Bayes	66	77	58	75	50	67
Bayes net	63	80	59	79	39	72
Multilayer perceptron	68	78	65	83	57	73

In Table 3, Thepade's SBTC is applied to the LUV color plane. For 70–30 training–testing data, Thepade's SBTC has a maximum accuracy of 60 and 62% in 10-ary using a logistic classifier and multilayer perceptron, respectively.

In Tables 4 and 5, Thepade's SBTC + LBP is used on the RGB color plane. Tables 4 and 5 show how Thepade's SBTC + LBP compares to the proposed technique. Using a multilayer perceptron, Thepade's SBTC + LBP achieves a maximum accuracy of 68% in 7-ary, while integrating Thepade's SBTC + LBP with GLCM achieves a greater accuracy of 76% in 4-ary and 75% in 7-ary using arndom forest 10-fold cross validation. Whereas Thepade's SBTC + LBP for 70–30 training–testing data has the greatest accuracy of 67% in 10-ary using a logistic classifier, combining Thepade's SBTC + LBP with GLCM yields greater accuracy of 75% for logistic classifier and 84% in random forest classifier in 7-ary.

In Table 6, the RGB color plane is subjected to GLCM feature extraction. For both ten-fold cross validation and 70–30 training–testing data using random forest classifier, the greatest accuracy for intel image classification dataset is 73%.

Table 3 Thepade's SBTC was applied on the LUV color plane in percentage

Algorithms	4-ary		7-ary		10-ary	
	10-fold cross validation	70/30 Training–testing	10-fold cross validation	70/30 Training–testing	10-fold cross validation	70/30 Training–testing
Random tree	46	51	48	49	52	47
Random forest	48	51	52	54	49	53
Simple logistic	55	53	57	58	56	56
Logistic	53	54	58	58	58	60
Naive Bayes	60	45	51	47	51	52
Bayes net	51	57	52	56	52	53
Multilayer perceptron	51	52	56	62	58	56

Table 4 Thepade's SBTC + LBP and Thepade's SBTC + LBP with GLCM applied on RGB color images for 10 fold cross validation in percentage

Algorithms	Thepade's SBTC 4-ary	Thepade's SBTC 4-ary + GLCM	Thepade's SBTC 7-ary	Thepade's SBTC 4-ary	Thepade's SBTC 10-ary	Thepade's SBTC 10-ary + GLCM
Random tree	50	63	54	64	57	65
Random forest	52	76	59	75	62	74
Simple logistic	57	72	56	71	63	71
Logistic	58	71	57	70	61	71
Naive Bayes	52	71	50	70	56	69
Bayes net	52	73	56	73	56	69
Multilayer perceptron	59	74	65	70	68	73

Table 5 Thepade's SBTC + LBP and Thepade's SBTC + LBP with GLCM applied on RGB color images for 70–30 training data in percentage

Algorithms	Thepade's SBTC 4-ary	Thepade's SBTC 4-ary + GLCM	Thepade's SBTC 7-ary	Thepade's SBTC 4-ary	Thepade's SBTC 10-ary	Thepade's SBTC 10-ary + GLCM
Random tree	48	63	55	69	60	66
Random forest	51	74	54	75	65	73
Simple logistic	60	70	59	69	57	71
Logistic	62	69	59	70	67	72
Naive Bayes	50	40	45	65	54	68
Bayes net	52	67	47	67	50	67
Multilayer perceptron	58	69	58	73	65	71

Table 6 Feature extracted using GLCM in percentage

Algorithms	10-fold cross validation	70/30 training–testing
Random tree	64	65
Random forest	73	73
Simple logistic	72	65
Logistic	69	66
Naive Bayes	68	59
Bayes net	72	69
Multilayer perceptron	73	66

In comparison with feature extraction utilizing Thepade's SBTC for RGB and LUV color planes, Thepade's SBTC + LBP, and Thepade's SBTC + LBP with GLCM; the fusion of Thepade's SBTC with GLCM for the RGB color plane has the highest accuracy of 83%.

6 Conclusion

Photographs have become a necessary part of human life. Images are being captured all across the world. One of the significant issues in scene classification is that the photographs are taken in a variety of locations. There are many diverse scenes in these locations, such as an airport, a restaurant, a cafeteria, and a forest. Indoor scene classification and outdoor scene classification are the most common types of scene categorization. Scene classification is challenging, which poses issues in research and image processing. Image classification becomes challenging because of the numerous distinct items present in various sceneries, causing a machine to become confused about the scene's categories. The current challenge with scene classification results in low-classification accuracy. The object comes in a variety of shapes and sizes and can be found in a variety of locations, which is one of the reasons for the scene classification's poor accuracy. We will perform feature extraction for pattern recognition and multiple machine learning models for image categorization using the intel image classification dataset. Thepade's SBTC *n*-ary feature extraction approach for RGB and LUV color plane models and fusion LBP and GLCM is presented in this study. When the three feature extraction methods fusion of Thepade's SBTC and GLCM with RGB color plane were compared, the fusion of Thepade's SBTC with GLCM for RGB color plane has the highest accuracy of 83%.

References

1. Thepade SD, Patil PH (2015) Novel video keyframe extraction using KPE vector quantization with assorted similarity measures in RGB and luv color spaces. In: 2015 international conference on industrial instrumentation and control (ICIC), pp 1603–1607. https://doi.org/10.1109/IIC.2015.7151006
2. Gill JS, Brar AS (2019) Support vector based indoor scene classification technique using different features. In: 2019 3rd international conference on electronics, communication and aeroplane technology (ICECA), Coimbatore, India, pp 685–689. https://doi.org/10.1109/ICECA.2019.8822153
3. Hayat M, Khan SH, Bennamoun M, An S (2016) A spatial layout and scale invariant feature representation for indoor scene classification. In: IEEE Trans Image Process 25(10):4829-4841. https://doi.org/10.1109/TIP.2016.2599292
4. Sun N, Zhu X, Liu J, Han G (2017) Indoor scene recognition based on deep learning and sparse representation. In: 2017 13th international conference on natural computation, fuzzy systems and knowledge discovery (ICNC-FSKD), Guilin, pp 844–849. https://doi.org/10.1109/FSKD.2017.8393385
5. Kabbai L, Abdellaoui M, Douik A (2019) Image classification by combining local and global features. Vis Comput 35:679–693. https://doi.org/10.1007/S00371-018-1503-0
6. Thepade SD, Gokhale A, Patki A, Khindkar J, Chaudhary P (2021) Arial image classification using deep neural networks with discrete cosine transform, TSBTC and augmentation techniques. international conference on emerging smart computing and informatics (ESCI), pp 396–401. https://doi.org/10.1109/ESCI50559.2021.9397010
7. Thepade SD, Bakshani DM, Bhingurde T, Burghate S, Deshmankar S (2020) Thepade's sorted block truncation coding applied on local binary patterns of images for splicing identification using machine learning classifiers. IEEE Bombay section signature conference (IBSSC), pp 208–213. https://doi.org/10.1109/IBSSC51096.2020.9332219
8. Thepade SD, Yadav NB (2015) Assessment of similarity measurement criteria in Thepade's sorted ternary block truncation coding (TSTBTC) for content based video retrieval. In: 2015 international conference on communication, information computing technology (ICCICT), pp 1–6. https://doi.org/10.1109/ICCICT.2015.7045728
9. Kekre HB, Thepade SD (2009) Colorization of grayscale images using Kekre's luv color space. In: Proceedigns of national level technical paper presentation competition thinkquest-2009, vol 14
10. Kekre HB, Thepade SD, Sanas SP (2010) Improved CBIR using multileveled block truncation coding. Int J Comput Sci Eng 8(2):2535–2544
11. Badre SR, Thepade SD (2016) Summarization with key frame extraction using Thepade's sorted *n*-ary block truncation coding applied on haar wavelet of video frame. In: Conference on advances in signal processing (CASP), pp 332–336. https://doi.org/10.1109/CASP.2016.7746190
12. Image classification dataset. https://www.kaggle.com/puneet6060/intel-image-classification

Analytical Study of Content-Based and Collaborative Filtering Methods for Recommender Systems

Srujan Putta and Omkaresh Kulkarni

Abstract Recommender systems (RSs) are an integral part of daily life. The main purpose of these systems is to suggest relevant items to the users and the use of recommender systems are critical in industries. RS is broadly classified into content-based filtering and collaborative filtering. This paper analyzes the aforementioned recommendation techniques. Content-based filtering system recommends relevant items to the user based on feedback or previous actions. Contrarily, collaborative filtering uses similarities between users or items to suggest relevant items. This study uses MovieLens 20 M dataset which has 20 million ratings and 465,000 tag applications applied to 27,000 movies by 138,000 users. The study analyzes the issue of content-based filtering in which the system is unable to suggest different genres to the user and primarily focuses on the user's interest which uses one-hot encoding approach and a dot product of the input user ratings vector and movies feature matrix for generating a user profile on which the system recommends suggestions to the user. Collaborative filtering technique overcomes this issue and provides robust suggestions which finds suggestions based on top neighbors, and similarity score between users is calculated by Pearson correlation method. Both the systems have advantages and disadvantages; content-based systems are unable to recommend movies of the genres which the user has not rated, which restricts the scope of recommendation. The paper analyzes the issue of limited exploration in content-based filtering and depicts how collaborative filtering provides generic solution.

Keywords Recommender systems · Content-based filtering · Collaborative filtering

S. Putta (✉) · O. Kulkarni
MIT-WPU, Pune, India
e-mail: srujanputta4@gmail.com

O. Kulkarni
e-mail: omkaresh.kulkarni@mitwpu.edu.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_44

1 Introduction

Technological progress has a huge impact on our lives; the way we sense, act, and think meeting needs at our finger tips. Recommender systems (RSs) are used extensively in industrial settings, especially in e-commerce, to suggest relevant items to the users. The goal of these systems is to improve the quality of prediction. Both the provider and customer benefit from these systems; for instance, the system suggests movie to customers based on their liking. Concomitantly, it is advantageous to the providers as it generates massive revenue. Based on the approach followed to recommend items to the users, RS is split into two main categories, namely content-based filtering (CBF) and collaborative filtering (CF).

1.1 Content-Based Filtering

CBF is based on the idea of building a model with the available features, which explains the observed user-item interactions, and answers the question ‘show me more of the same of what I have liked before’. A user profile is built, based on which similar items are suggested to the user. To build a user profile, one-hot encoding approach is followed. Later, the vector is multiplied with the feature matrix to find the weighted feature matrix; once created, the weighted feature matrix is aggregated and normalized, which is then used for prediction. Here, the vector is the input user rating and the movies matrix including movie and genre is the feature matrix.

Figure 1 depicts the principle of CBF, where relevant items are suggested. The user likes item ‘A’ and item ‘B’. Further, the user profile generated by one-hot encoding and dot product of input user rating and the movies matrix, weighted movies matrix is created. Here, item ‘C’ closely matches with item ‘A’, hence item ‘C’ is suggested to the user. Whereas, item ‘D’ does not match with the user profile, hence the item is not recommended.

1.2 Collaborative Filtering

CF approach uses different techniques to find similarities between users or items, to provide suggestions. It answers the question, ‘Tell me what’s popular among my neighbors’. There exists a relationship between product and people’s interest. Later, CF is bifurcated into user based and item based; user based is a technique where the recommendations are focused on the neighbors of the user, whereas item based is dependent on neighboring items. This paper is concerned with user-based CF with movies dataset. Based on the user’s movie ratings, top-n neighbors are found. Then, Pearson correlation is used as a statistical algorithm to find the similarity index

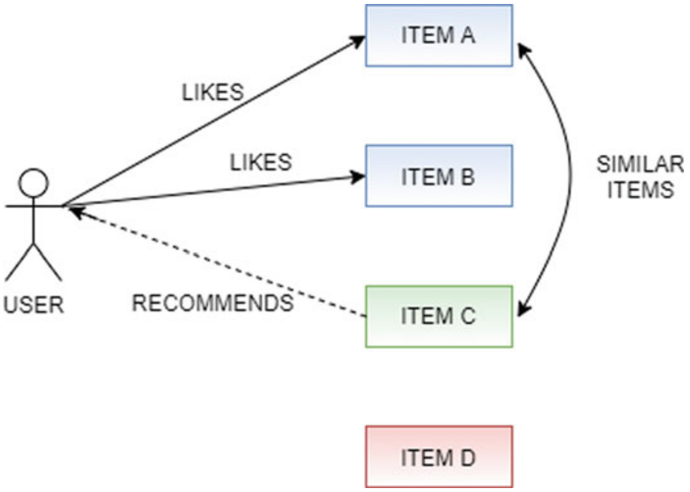


Fig. 1 Principle of content-based filtering

between users, which is then used to suggest movies with the highest score to the user.

Figure 2 illustrates the principle of CF where the items are recommended based on the neighbors, and it uses Pearson correlation as a similarity matrix. Here, users ‘1’ and ‘2’ both like items ‘A’ and ‘B’. Later, user ‘2’ like item ‘C’, since users ‘1’ and ‘2’ had high similarity score as they liked same movies, item ‘C’ is recommended to user ‘1’.

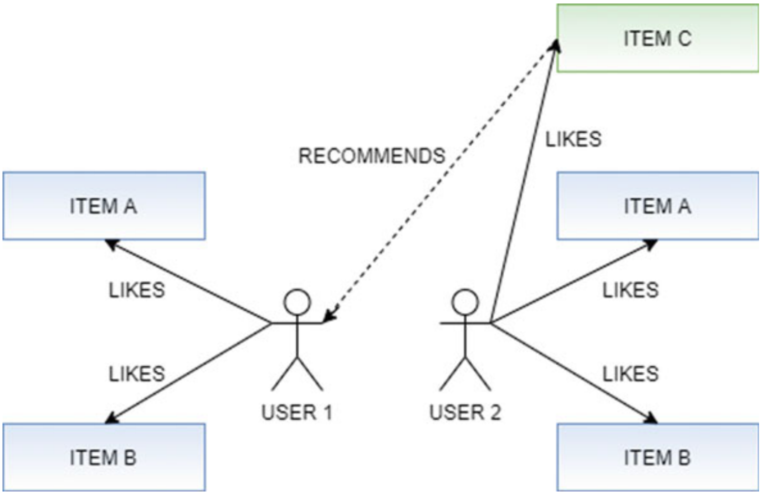


Fig. 2 Principle of collaborative filtering

The CBF method uses a user profile for recommending items, but the main drawback of CBF is that it hardly suggests new items to the user. In this paper, we discuss the situation where movies of genres similar to those of the user profile are recommended to users and newer predictions are sparse. On the contrary, the CF method makes recommendations based on neighborhood; therefore, solving this issue and expanding users' interest by suggesting movies of different genre based on what the adjacent of the user like. The resulting bar graph shows the distinguishing factor between CBF and CF.

2 Related Work

Recommender systems are used widely in the industry and are effective in providing relevant suggestions. Movie recommendation is one of the prominent applications of RS which uses CBF, CF, or hybrid approach for recommendation.

Music recommendations by content-based method music subjective features such as speech, volume, and acoustics are analyzed in paper by Pal et al. [1]. In content-based method, whenever user fires a query to database music feature attribute value compares with clusters centroid, whereas, in the collaborative method, ratings given by users to particular music is considered and adjusted cosine similarity is used to find similarity between users mentioning cold-start problem for a new user.

The paper [4] analyzed and compared the performance of the collaborative filtering and hybrid-based approaches in generating movie recommendations. Hybrid uses both rating and movie data and consists of four main phases: (1) text preprocessing, (2) term weighting, (3) movie clustering, and (4) collaborative filtering-based approach. The findings conjectured that hybrid approach does not always improve the collaborative filtering approach in movie recommendation.

Collaborative filtering can be classified into user based and item based; these techniques have both advantages and disadvantages. Gupta and Katarya in paper [5] analyzed the two techniques on a benchmark MovieLens dataset, and the resultant findings indicate the performance of each algorithm as well as the algorithm that provides better results; item-based collaborative filtering is efficient than user-based collaborative filtering.

According to paper [6], the RS provides property information based on user behavior by searching advertising content previously searched by the user. The application system presents the same product recommendation in accordance with the profile/criteria and preference of the prospective buyer. Therefore, the RS assists prospective buyers in determining the choice of property product they want to buy and this process can be provided by the RS in a short time.

Book RS is based on the combined features of CBF, CF, and association rule mining to produce efficient and effective recommendations. In paper [8], Mathew et al. proposed a hybrid, combining two or more algorithms which facilitates the RS in suggesting the book based on the buyer's interest.

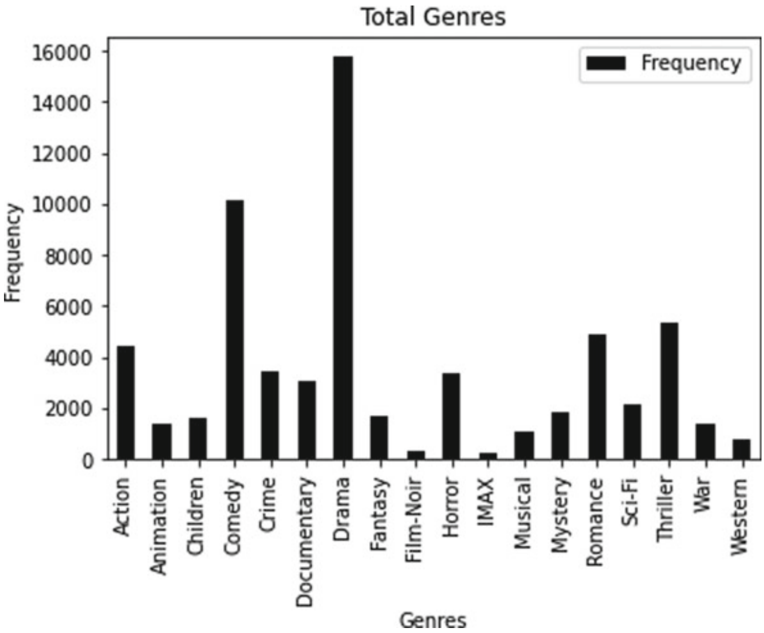


Fig. 3 Frequency count of total genres in the dataset

3 Proposed Methodology

3.1 Dataset

The data used for analysis is MovieLens 20 M dataset provided by GroupLens research of size 190 MB. Movie title, movie ID, genres, ratings from different users on the scale of 0 to 5, and user ID are the features of our dataset. It consists of 34,208 movies with ratings from 1,38,00 of users, tallying ratings to 2,28,84,377 values from year 1995 to 2015. There are a total of 18 genres; action, animation, children, comedy, crime, documentary, drama, fantasy, film-noir, horror, IMAX, musical, mystery, romance, sci-fi, thriller, war, and western. Most of the movies have multiple genres, and drama is the most prominent with a frequency of 15,774, followed by comedy, with 10,124 frequency count as shown in Fig. 3.

3.2 Content-Based Filtering

The use of user profile governs the CBF approach. First, libraries required for analysis are imported. Later, the data is preprocessed to get it in the required format. The user input is accepted based on which the profile is created. For instance, we have

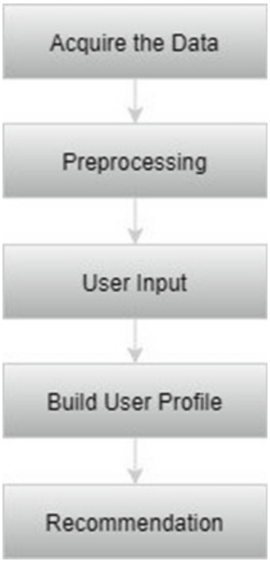


Fig. 4 Process of CBF

considered 14 movies as an input from the user along with ratings of the movies. Further, to build a user profile, one-hot encoding methodology is used; new categorical columns are created and binary values 0 and 1 are assigned to categorical values in the column. All the values are zero except one, which indicates the index. This makes the data expressive and can be rescaled whenever required.

The input user ratings are a vector, and the movies feature matrix is generated by one-hot encoding. The weighted genre matrix is a dot product of the input user ratings vector and movies feature matrix. Furthermore, the weighted feature matrix of genres is aggregated and then normalized to find the user profile. The genres are then multiplied by the user profile, and the weighted average is taken to form a recommendation table. In this study, we recommend the top 50 movies to the user, based on the profile. Figure 4 describes process of CBF, where the data is acquired form MovieLens 20 M dataset and is preprocessed. Here, the movie title and year of publication are separated into two separate columns from the original dataset. Moreover, the genres are converted to list and further into columns for one-hot encoding method. Later, user input is accepted and weighted user rating is built by dot product and one-hot encoding which is used for recommending movies to the users.

3.3 Collaborative Filtering

CF is classified into two types: item based and user based. This study focuses on user-based CF, which finds users that have similar choices as the input and recommends items, that neighboring users have liked, to the input. Based on the user's adjacent users, relevant items are suggested by calculating the similarity score between multiple users. The primary step involved importing necessary libraries and preprocessing the dataset, and later, we used the same user input that was used for CBF. In CF, we take into account the user ID, movie ID, and ratings of the movies. Additionally, users who have watched the same movies as the input user are clubbed together and sorted, based on the movies most in common. We arranged these groups so the users that have most movies common with the input have higher priority; it provides a richer recommendation since it is not possible to go through every single user. A similarity score is calculated based on the ratings. 'Pearson correlation', 'adjusted cosine similarity', and 'Euclidean distance' are widely used for calculating the similarity between users.

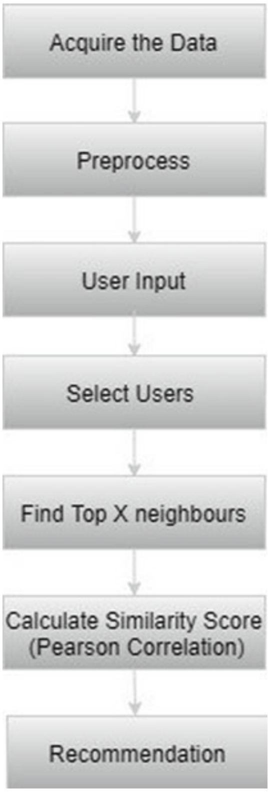
We determined the similarity between each user and the input through the Pearson correlation coefficient. It is used to measure the strength of a linear association between two variables; the Pearson correlation similarity is invariant to scaling. For instance, for two vectors $V1$ and $V2$, the Pearson correlation function is defined as: Pearson ($V1$, $V2$) = Pearson ($V1$, 2 * $V2$ + 3). In this study, the Pearson correlation was stored in a dictionary, where the key is the user ID and the value is the coefficient and calculated for every user group in our subset. After calculating the similarity index for each user, we multiplied the similarity by user's ratings to generate weighted ratings. Later, it is aggregated and weighted average recommendation score is calculated; thus, the top 50 movies to the user based on its neighborhood are represented.

The formula for determining the Pearson correlation coefficient between sets X and Y with N values is

$$r = \frac{\sum (x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum (x_i - \overline{x})^2 \sum (y_i - \overline{y})^2}}$$

In the formula, r represents correlation coefficient, $\overline{x}$ and $\overline{y}$ represent mean of the values of the x-variable and y-variable, respectively, and xi and yi represent the values of x- and y-variables in a sample. The values given by the formula vary from $r = -1$ to $r = 1$, where 1 forms a direct correlation between the two entities which indicates a perfect positive correlation and -1 indicates a perfect negative correlation. In this study, a 1 means that the two users have similar tastes, while a -1 means the opposite. Figure 5 shows the process of CF, where firstly, the data is taken from MovieLens 20 M dataset, and similar preprocessing is done as for CBF, and movie ratings are taken into consideration. Further, the user input accepted is used to generate top X neighbors and similarity score; namely, Pearson correlation is used to accomplish this which is then used for recommendations.

Fig. 5 Process of collaborative filtering



4 Results

We compared the two approaches for RS indicating the issue of predictions based on limited interests, which restricts the scope of a user in CBF. CF uses user's neighbors to find the recommendations, which solves the disadvantage faced in CBF.

In CBF, first, a user profile is generated based on the inputs and later it is used for suggestions. The inputs considered for demonstration comprised 14 movies and the corresponding ratings. Table 1 represents users input which consists of movie title, genres, and movie ratings based on which the user profile is created for CBF and weighted average recommendation score is calculated for CF by finding similarity score. Further, Tables 2 and 3 represent the recommendations from both recommender systems.

Table 2 represents the recommendations from CBF, where the movies having similar genres to the user profile are suggested. Here, the user finds movies which are of the same taste, i.e., action, adventure, drama, and thriller, and new genre movies are not suggested, which is deleterious for the service provider as well as the customer as it refrains the user from exploring new genres. On the other hand, CF provides flavorful recommendations as shown in Table 3, so that the users can have

Table 1 User input—movie title, genres, and ratings

Movie title	Genres	Rating
Senna	Documentary	1.5
Iron man	Action, adventure, sci-fi	5
Iron man 2	Action, adventure, sci-fi, thriller, IMAX	4
Guns at Batasi	Drama	3
Swamp water	Drama	3
Suspicious river	Drama	5
Hot coffee	Documentary	4
Casino	Crime, drama	5
Across the sea of time	Documentary, IMAX	2.5
When night is falling	Drama, romance	3
Braveheart	Action, drama, war	2
Congo	Action, adventure, mystery, sci-fi	1
Blood in the face	Documentary	4.5
Methadonia	Documentary	5

Table 2 Top ten recommendations—content-based filtering

Movie title	Genres
RoboCop 3	Action, crime, drama, sci-fi, thriller
Blood diamond	Action, adventure, crime, drama, thriller, war
The Bourne legacy	Action, adventure, drama, thriller, IMAX
The wreaking crew	Action, adventure, comedy, crime, drama, thriller
Sanctum	Action, adventure, drama, thriller, IMAX
The hero: love story of a spy	Action, adventure, drama, musical, romance, thriller
Mobile suit Gundam II	Action, adventure, animation, crime, drama, sci-fi, thriller
City of god	Action, adventure, crime, drama, thriller
humanity’s end	Action, adventure, drama, sci-fi, thriller
Ghosts of cite Soleil	Action, documentary, drama, romance, war

unhindered access to a number of genres which are likely to have positive responses from the user as these movies are suggested on the basis of user-based filtering which incorporates user’s neighbors.

Figure 6 shows the frequency count of genres based on user profile and inputs from Table 1. Based on the profile, it suggests that users preferred to watch drama the most, followed by documentaries, action, and adventure, while sci-fi, thriller, and crime had moderate preference. The users did not watch movies belonging to the following genre: animation, children, comedy, fantasy, film-noir, horror, musical, and western.

Table 3 Top ten recommendations—collaborative filtering

Movie title	Genres
Who is Harry Nilsson	Documentary
The house of small cubes	Animation, drama
Sweet movie	Comedy, drama
Patton Oswalt: finest hour	Comedy
O Auto da Compadecida	Adventure, comedy
Othello	Drama
The jinx: the life and deaths of Robert durst	Documentary
Amy	Documentary
Jiminy Glick in La La Wood	Comedy, mystery
Patton Oswalt: werewolves and lollipops	Comedy

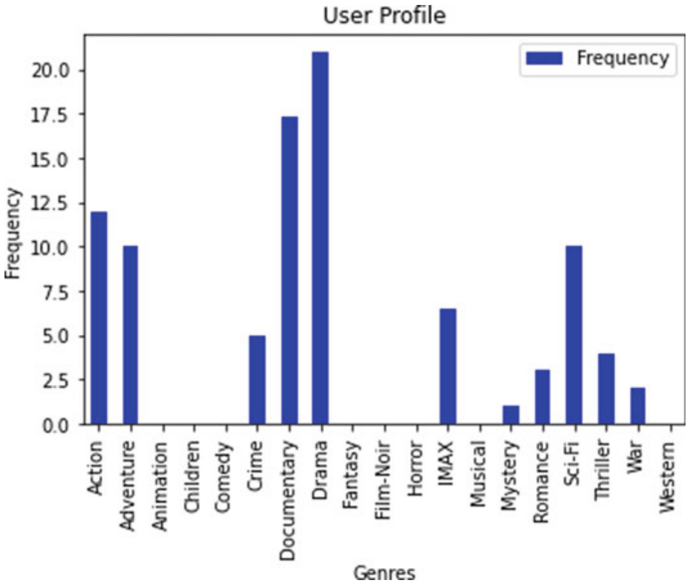


Fig. 6 Frequency count of genres based on user profile

Once the user profile is generated, CBF is implemented and movies are suggested to the user. Figure 7 depicts the frequency of genres recommended in the top 50 suggestions. The resulting bar graph of CBF shares similar genre density with the user profile graph. It provides highly customized recommendations, but lacks exploration. The frequency was higher for action, adventure, drama, documentary, and sci-fi movies. As the user did not watch animation, children, comedy, fantasy, film-noir, musical, and western, hardly any movies belonging to these genres were recommended, which limits the interests of users. This shuts the door for exploring movies

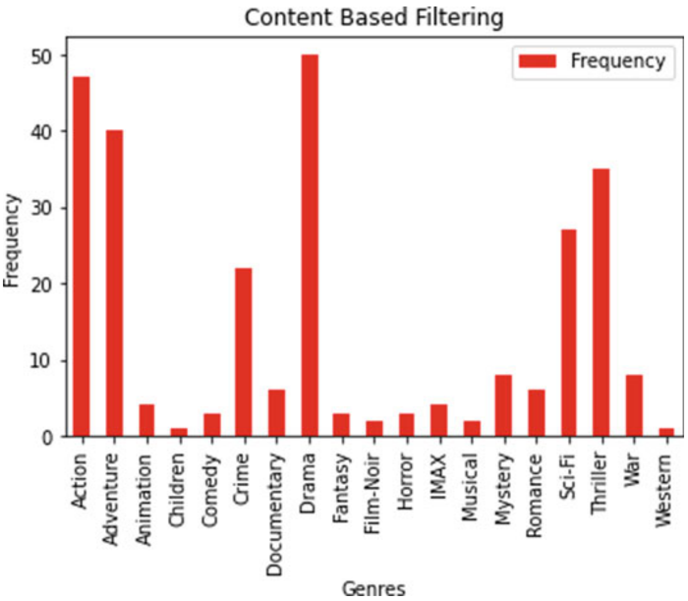


Fig. 7 Frequency count of genres based on CBF

of other genre, thus depicting limited ability to expand users' interest. For instance, if the user has watched only few movies, the recommendations provided will have a limited scope and less precision, as hardly any movies of other genre will be suggested.

CF utilizes the similarity between users to recommend movies to the user based on movies watched and liked by neighboring users. It ensures that no one genre is suggested to the user, but overall recommendation is provided based on the movies which are highly rated by its neighboring users. The bar graph in Fig. 8 shows genre frequency of top 50 recommended movies. No domain knowledge is required for CF, and users can discover new interests. It adapts to the users' interest which may change over a period of time. Contrarily, CF faces the issue of 'cold-start' where the model fails to recommend items to new users. In addition, privacy issue arises when trying to learn users' preferences.

5 Conclusion

Movie recommendation is a prime application of RS. This study analyzes two approaches for RS, namely CBF and CF. CBF uses a user profile to suggest movies, whereas the latter calculates the similarity between users based on Pearson correlation. CBF restricts the exploration of the users by suggesting movies of specific genres based on the user profile, whereas CF provides robust suggestions to the users

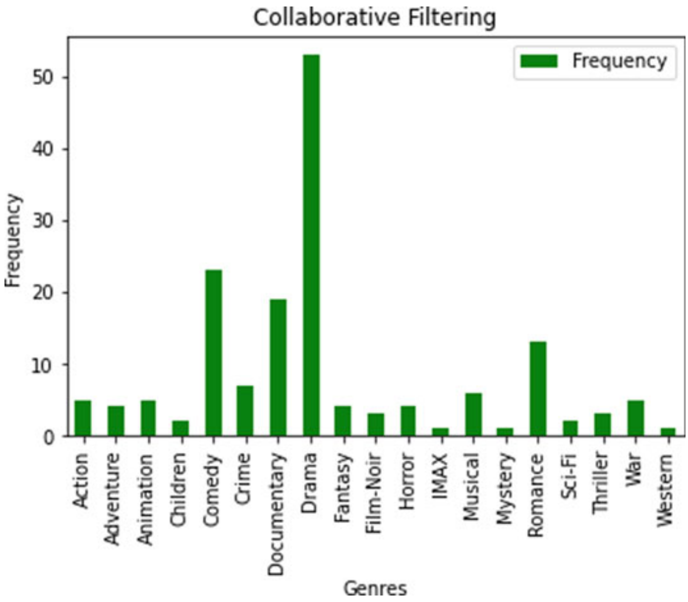


Fig. 8 Frequency count of genres based on collaborative filtering

by taking adjacent users into consideration. The paper highlights the disadvantage of CBF; which limits the scope of users' preferences, and the experimental results show how CF overcomes this issue.

For future research, we plan to work on hybrid recommender systems, combining the two approaches CBF and CF for improving the quality of recommendations and overcoming the shortcomings of current systems. Moreover, for larger datasets, we plan on using Hadoop distributed systems for efficient performance.

References

1. Darshna P (2018)Music recommendation based on content and collaborative approach and reducing cold start problem. In: 2018 2nd international conference on inventive systems and control (ICISC), pp 1033–1037. https://doi.org/10.1109/ICISC.2018.8398959
2. Pal A, Parhi P, Aggarwal M (2017)An improved content based collaborative filtering algorithm for movie recommendations. In: 2017 10th international conference on contemporary computing (IC3), pp 1–3. https://doi.org/10.1109/IC3.2017.8284357
3. Zhao L, Pan Z (2021)Research on online course recommendation model based on improved collaborative filtering algorithm. In: 2021 IEEE 6th international conference on cloud computing and big data analytics (ICCCBDA), pp 437–440. https://doi.org/10.1109/ICCCBDA51879.2021.9442575
4. Ifada N, Rahman TF, Sophan MK (2020)Comparing collaborative filtering and hybrid based approaches for movie recommendation. In: 2020 6th information technology international seminar (ITIS), pp 219–223. https://doi.org/10.1109/ITIS50118.2020.9321014

5. Gupta G, Katarya R (2019) Recommendation analysis on item-based and user-based collaborative filtering. Int Conf Smart Syst Invent Technol (ICSSIT) 2019:1–4. https://doi.org/10.1109/ICSSIT46314.2019.8987745
6. Badriyah T, Azvy S, Yuwono W, Syarif I (2018) Recommendation system for property search using content based filtering method. Int Conf Inform Commun Technol (ICOIACT) 2018:25–29. https://doi.org/10.1109/ICOIACT.2018.8350801
7. Hornung T, Ziegler C, Franz S, Przyjaciel-Zablocki M, Schätzle A, Lausen G (2019)Evaluating hybrid music recommender systems. In: 2013 IEEE/WIC/ACM international joint conferences on web intelligence (WI) and intelligent agent technologies (IAT), pp 57–64. https://doi.org/10.1109/WI-IAT.2013.9.G. Gupta, Katarya R (2019) Recommendation analysis on item-based and user-based collaborative filtering. In: 2019 International conference on smart systems and inventive technology (ICSSIT), pp 1–4. https://doi.org/10.1109/ICSSIT46314.2019.8987745
8. Mathew P, Kuriakose B, Hegde V (2016) Book recommendation system through content based and collaborative filtering method. Int Conf Data Mining Adv Comput (SAPIENCE) 2016:47–52. https://doi.org/10.1109/SAPIENCE.2016.7684
9. Maxwell Harper F, Konstan JA (2015) The MovieLens datasets: history and context. ACM Trans Interact Intell Syst (TiiS) 5, 4(19):19. https://doi.org/10.1145/2827872
10. Hartin T, Williams Y (2021) Process of finding positive correlation. Retrieved from https://study.com

Emotion Detection Using Facial Expressions

Shivani Nandani, Rohin Nanavati, and Manish Khare

Abstract Emotion detection has been an area of interest for researchers for several decades. Since emotions can influence our decisions in life, they have been defined as an essential part of every human behaviour predictor. With the increasing need to understand human behaviour, facial expressions are vital in recognising emotions via non-verbal communication. For this study, two methods were used—CNN and PCA + SVM—with two data sets—the CK+ data set and the JAFFE data set. This study intends to compare some of the widely accepted machine learning algorithms for the given problem and a given data set.

Keywords Emotion detection · Facial expressions · CNN · PCA · SVM

1 Introduction

Emotions help us convey our intentions and form an essential part of communication that requires no language. They are universal. Our facial expressions often betray the emotions we are feeling. According to Mehrabian [12], 55% of a message pertaining to feeling and attitude is in the facial expression.

Initially studied by Ekman [3], facial features can be classified into six basic emotions—happiness, sadness, fear, disgust, surprise and anger. Therefore, emotion detection is possible via facial expressions, called facial emotion recognition (FER). FER has significant academic and commercial potential.

Our objective is to detect emotions using facial expressions to predict emotions. The scope of this paper remains limited to FER, whilst we continue to work on trans-

S. Nandani · R. Nanavati · M. Khare (✉)
Dhirubhai Ambani Institute of Information and Communication Technology, Gandhinagar, India
e-mail: manish_khare@daiict.ac.in

S. Nandani
e-mail: 201801076@daiict.ac.in

R. Nanavati
e-mail: 201801108@daiict.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_45

lating the results into real-time applications. Significant research has been conducted in the field of FER. After reviewing the current state of the art methods, we chose to experiment using two different approaches. We attempted FER using a convolution neural network (CNN) as our choice of unsupervised learning method. For supervised learning, we chose to use a support vector machine (SVM) to solve the classification problem. The two data sets that we used were the JAFFE data set [10, 11] and the CK+ data set [9].

2 Related Work

Since emotions play an essential role in the choices that humans make in everyday life (such as what to eat or whom to talk to) as well as plan the future course of action (such as what career to pursue or whom to marry), the ability to correctly recognise emotions, or FER, is an area of interest for several researchers working in the domain of computer vision. The reason for this interest is the application of FER in several disciplines:

- In social interaction systems, facial expressions are an essential form of communication since they allow individuals to communicate their feelings in a non-verbal manner.
- Emotion processing is a crucial part of normal brain development in humans. Thus, tracking facial expressions from childhood to adolescence may prove helpful in understanding a child's growth [4].
- With the increase in AI-oriented technology and robotics, FER will be essential in these automated systems, especially when used to provide services that require a good analysis of the service user's emotional state. Examples of such scenarios would be security, entertainment, household and medicine [1].
- For distance learning programmes, which have become increasingly popular due to the COVID-19 pandemic, FER can identify students' understanding during the study process and, subsequently, change teaching strategies based on the data obtained [15].

2.1 Convolution Neural Network

Krizhevsky and Hinton gave a landmark publication that showed how a deep neural network works and its resemblance to the human visual cortex's functionality [7]. With the recent advances in deep learning, CNN models have been employed for several machine learning problems and have outperformed the existing models. In the case of FER, a typical model consists of three different stages—facial recognition, feature extraction and classification. CNN models are biologically inspired and thus

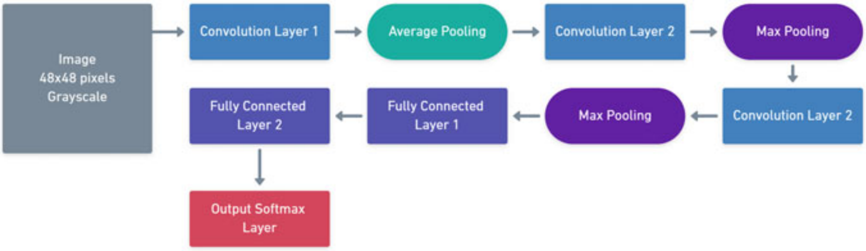


Fig. 1 The architecture proposed by Dachapally [2]

combine the feature extraction and classification steps. The input to a CNN model is a localised face image, and the output is classified labels. Several models have been proposed earlier—Liu et al. [8] proposed a model that uses three CNN subnets and achieved an accuracy of 65.03%; Dachapally [2] presented a model that used multiple convolution layers and achieved an accuracy of 86.38%. Shin et al. [14] proposed several other architectures that achieve ≈ 60% accuracy. The architecture used by Dachapally [2] is shown in Fig. 1.

2.2 Support Vector Machines

SVM is one of the oldest machine learning models and is still relevant in the industry. This can be attributed to the simple interpretation of an SVM model—it finds a hyperplane in an N-dimensional space that distinctly classifies data points.

SVM has been employed for the FER problem in the past. In the model proposed by Abdulrahman and Eleyan [1], the PCA + SVM model has been used on JAFFE and MUFE data sets and produces an average accuracy of 87% and 77%, respectively.

3 Data Sets

The data sets used for this study include the extended CK+ data set [9] and the JAFFE data set [10, 11]. Both data sets were divided into an 80% training set and a 20% validation set.

The **Extended Cohn-Kanade Data set (CK+)** [9] is a public benchmark data set for emotion recognition. It comprises a total of 327 labelled grey-scale images which have similar backgrounds. A sample of the same is shown in Fig. 2.

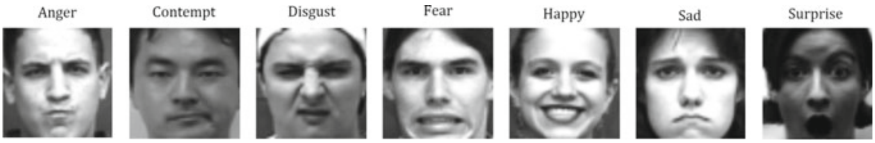


Fig. 2 Sample images—CK+ data set [9]

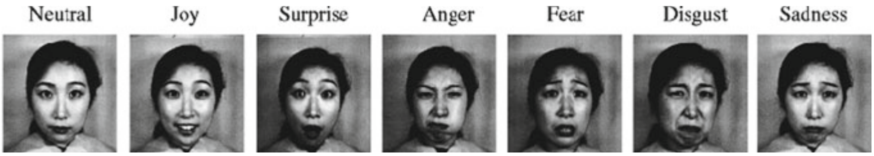


Fig. 3 Sample images—JAFFE data set [10, 11]

The **Japanese Female Facial Expression Data set (JAFFE)** [10, 11] has 213 labelled images. As shown in sample images in Fig. 3, all images are 8-bit grey scale and 256×256 pixels. 10 Japanese female expressers make up the data set with seven posed facial expressions (6 basic facial expressions + 1 neutral). This data set does not portray 'contempt'.

4 Proposed Method

Two different methods were used—convolution neural networks and principal component analysis combined with support vector machine.

4.1 Convolution Neural Networks

Based on the success of the previous publications with CNN-based models, we decided to build our model from scratch with four convolution layers, one pooling layer and one fully connected layer. The architecture is represented in Fig. 4.

Here, the input layer takes grey-scale images in 48 × 48 pixel format. The first, second, third and fourth convolution layers use 6 × 6 kernel size, 2 × 2 kernel size, 2 × 2 kernel size and 2 × 2 kernel size, respectively. A max-pooling layer follows the third convolution layer. For the max-pooling layers, we have used a 2 × 2 window. Finally, we have the fully connected layer and the output layer, which uses the 'softmax' function.

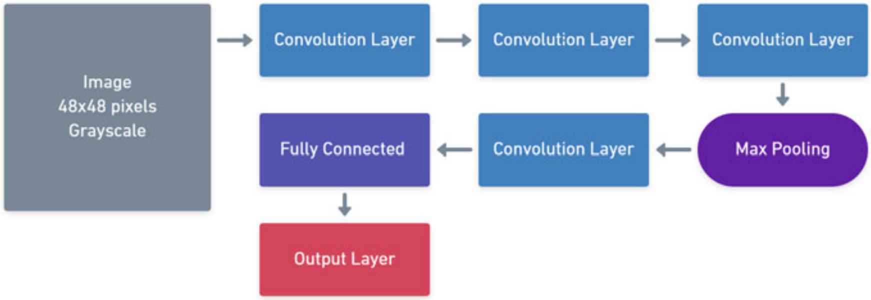


Fig. 4 Proposed CNN model

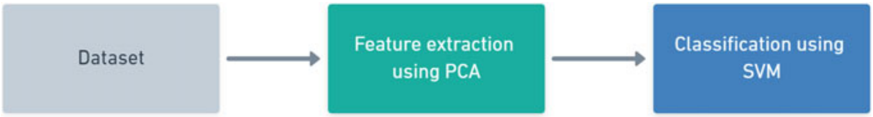


Fig. 5 Proposed PCA + SVM model

4.2 *Support Vector Machine (SVM)*

As seen in Fig. 5, we used principal component analysis (PCA) using randomised singular value decomposition for feature extraction. PCA reduces the image matrix into its principal vectors, sorted by the eigenvalues. We used the first 120–130 principal components for the SVM.

5 Experimental Results and Analysis

All the experiments were performed on the system with the hardware architecture shown in Table 1.

Table 1 Hardware architecture

Architecture	x86_64
Model name	Intel(R) Core(TM) i7-8750H CPU@2.20 GHz
Socket(s)	1
L1d cache	192 KiB
L2 cache	1.5 MiB
L3 cache	9 MiB

For both the methods mentioned in Sect. 4, we divided the data sets (CK+ [9] and JAFFE [10, 11]) into 80% training and 20% testing set. To evaluate the models, we have used accuracy as the performance metric.

5.1 CK+ Data Set

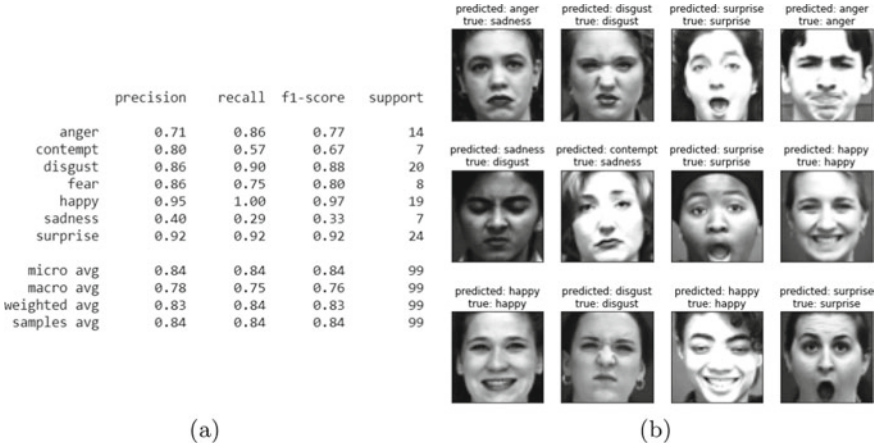
Both the models achieve an accuracy of above 80%. However, since the data set is skewed, some emotions are identified more accurately than others.

Convolution Neural Network For the CK+ data set, the CNN model gives 83.84% accuracy. The training was done for 80 epochs.

As seen in Fig. 6, the model is able to identify 'happy', 'surprise', 'fear' and 'disgust' with maximum precision, which is in accordance with human-level emotion detection. Some examples of the same are shown in Fig. 6. The model misclassifies 'sadness' the most. This misclassification results in an accuracy of 83.84%. The model loss and accuracy for each epoch can be seen in the graphs given in Fig. 7.

Support Vector Machine For the PCA + SVM model, we get an accuracy of 81.81%. We used the first 120 principal components.

As can be seen from Fig. 8, the significance of the principal components decreases rapidly. In fact, after the 14th component, the significance dropped below 1%. As we wanted to retain at least 95% of the image information, we needed to use more than 80 components, as can be observed from Fig. 8. With this constraint, we then chose 120 principal components as that gave the best accuracy.



```
              precision    recall  f1-score   support

       anger       0.71      0.86      0.77        14
    contempt       0.80      0.57      0.67         7
     disgust       0.86      0.90      0.88        20
        fear       0.86      0.75      0.80         8
       happy       0.95      1.00      0.97        19
     sadness       0.40      0.29      0.33         7
    surprise       0.92      0.92      0.92        24

   micro avg       0.84      0.84      0.84        99
   macro avg       0.78      0.75      0.76        99
weighted avg       0.83      0.84      0.83        99
 samples avg       0.84      0.84      0.84        99
```

predicted: anger
true: sadness
predicted: disgust
true: disgust
predicted: surprise
true: surprise
predicted: anger
true: anger
predicted: sadness
true: disgust
predicted: contempt
true: sadness
predicted: surprise
true: surprise
predicted: happy
true: happy
predicted: happy
true: happy
predicted: disgust
true: disgust
predicted: happy
true: happy
predicted: surprise
true: surprise
(a)
(b)

Fig. 6 Results for CNN model for CK+ data set [9] **a** classification report, **b** sample predictions

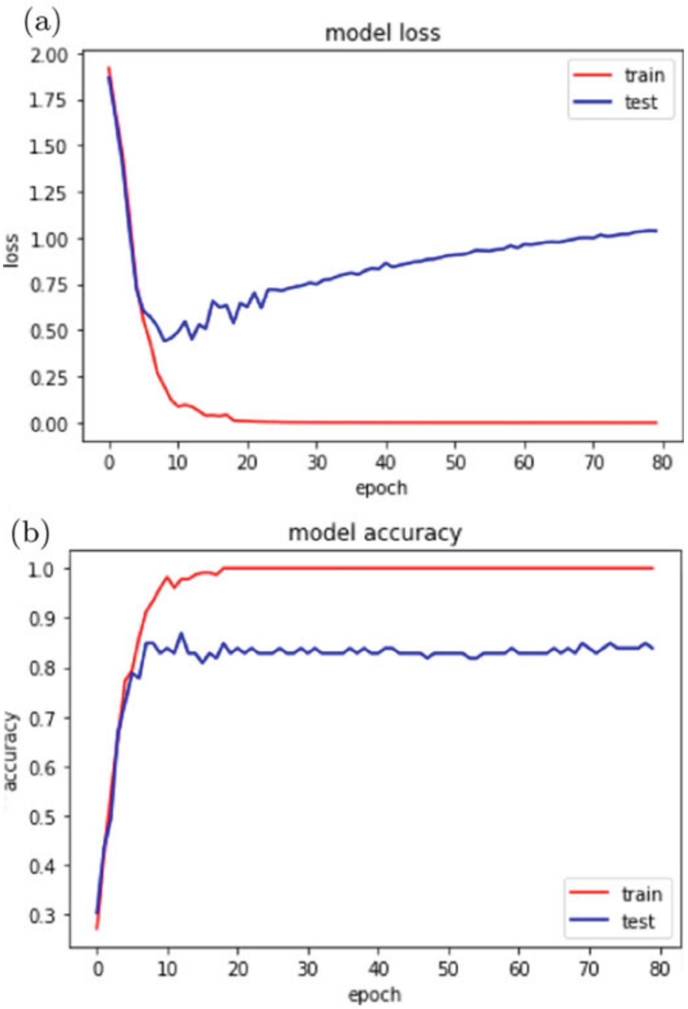


Fig. 7 **a** Model loss versus number of epochs, **b** model accuracy versus number of epochs

From the classification report given in Fig. 9, it is evident that the model, same as humans, can predict emotions such as 'happy', 'fear' and 'surprise' with greater accuracy than the other emotions. Some examples are shown in Fig. 9. 'Sadness' and 'anger' emotions are again misclassified the most, as was the case with the CNN model.

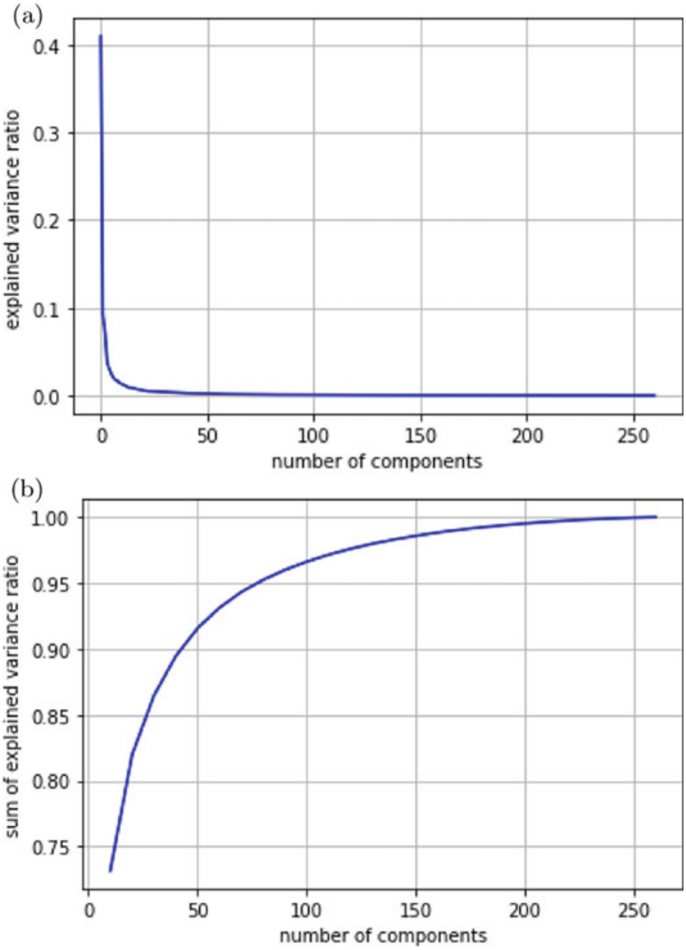


Fig. 8 **a** Explained variance ratio, **b** sum of explained variance ratio

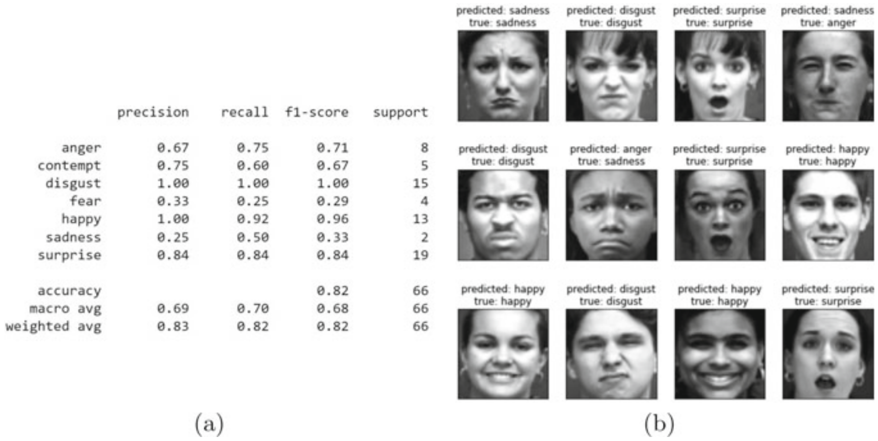
5.2 JAFFE Data Set

Both the models give accuracy above 85%; however, the PCA + SVM model beats the CNN model on the JAFFE data set [10, 11].

Convolution Neural Network When CNN is performed on the JAFFE data set, we get an accuracy of 87.50%. The training was done for 80 epochs.

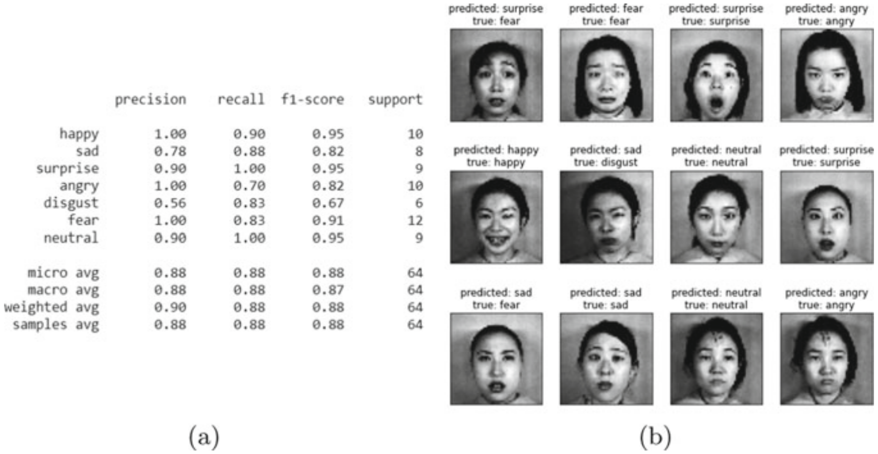
As seen in Fig. 10, the model can identify 'happiness', 'anger' and 'fear' with extremely high precision, whilst 'disgust' has the least precision value. Examples of the model's predictions are shown in Fig. 10.

The model loss and accuracy for each epoch can be seen in the graphs shown in Fig. 11.



```
              precision    recall  f1-score   support

       anger       0.67      0.75      0.71         8
    contempt       0.75      0.60      0.67         5
     disgust       1.00      1.00      1.00        15
        fear       0.33      0.25      0.29         4
       happy       1.00      0.92      0.96        13
     sadness       0.25      0.50      0.33         2
    surprise       0.84      0.84      0.84        19

    accuracy                           0.82        66
   macro avg       0.69      0.70      0.68        66
weighted avg       0.83      0.82      0.82        66
```

Fig. 9 Results for PCA + SVM model for CK+ data set [9] **a** classification report, **b** sample predictions



```
              precision    recall  f1-score   support

       happy       1.00      0.90      0.95        10
         sad       0.78      0.88      0.82         8
    surprise       0.90      1.00      0.95         9
       angry       1.00      0.70      0.82        10
     disgust       0.56      0.83      0.67         6
        fear       1.00      0.83      0.91        12
     neutral       0.90      1.00      0.95         9

   micro avg       0.88      0.88      0.88        64
   macro avg       0.88      0.88      0.87        64
weighted avg       0.90      0.88      0.88        64
 samples avg       0.88      0.88      0.88        64
```

Fig. 10 Results for CNN model for JAFFE data set [10, 11] (a) Classification report (b) Sample predictions

Support Vector Machine For the PCA + SVM model, we get an accuracy of 95.35%. We used the first 130 principal components.

As can be seen from Fig. 12, the significance of the principal components decreases rapidly. In fact, after the 15th component, the significance dropped below 1%. We wanted to retain at least 95% of the image information; therefore, we needed to use more than 80 components, as can be observed from Fig. 12. With this constraint, we then chose 130 principal components as that gave the best accuracy.

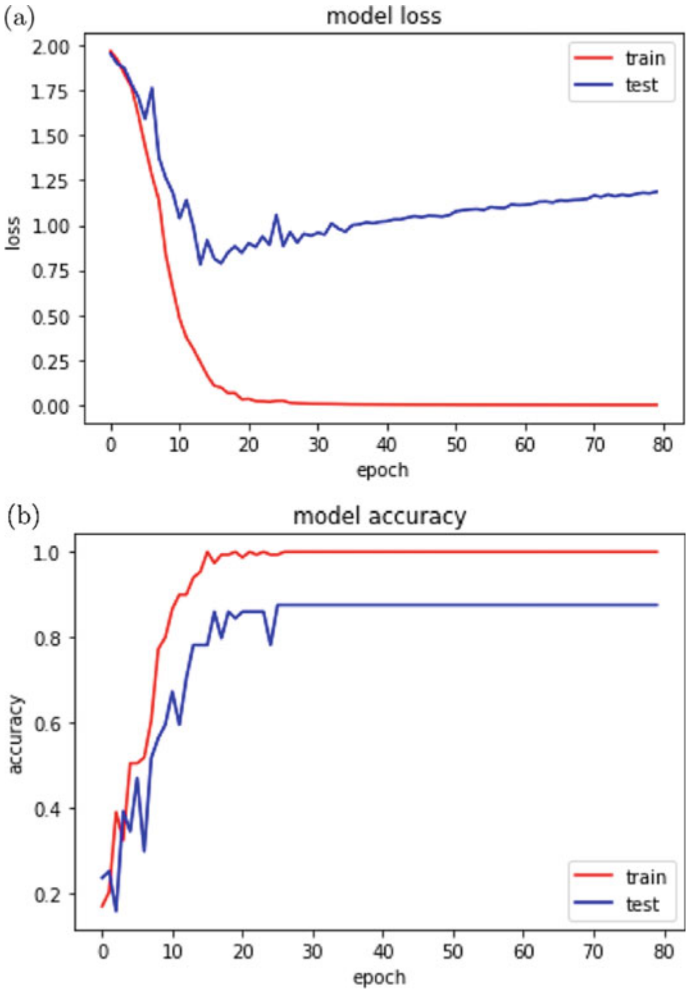


Fig. 11 **a** Model loss versus number of epochs, **b** model accuracy versus number of epochs

The model only struggled with 'neutral' and 'fear' emotions, as can be interpreted from Fig. 13. The high accuracy of the model is evidenced by the few examples shown in Fig. 13.

5.3 Against Other Comparable Architecture

Our proposed PCA + SVM model performs best on the JAFFE data set [10, 11] and beats several other proposed methods. As can be seen from Table 2, many methods

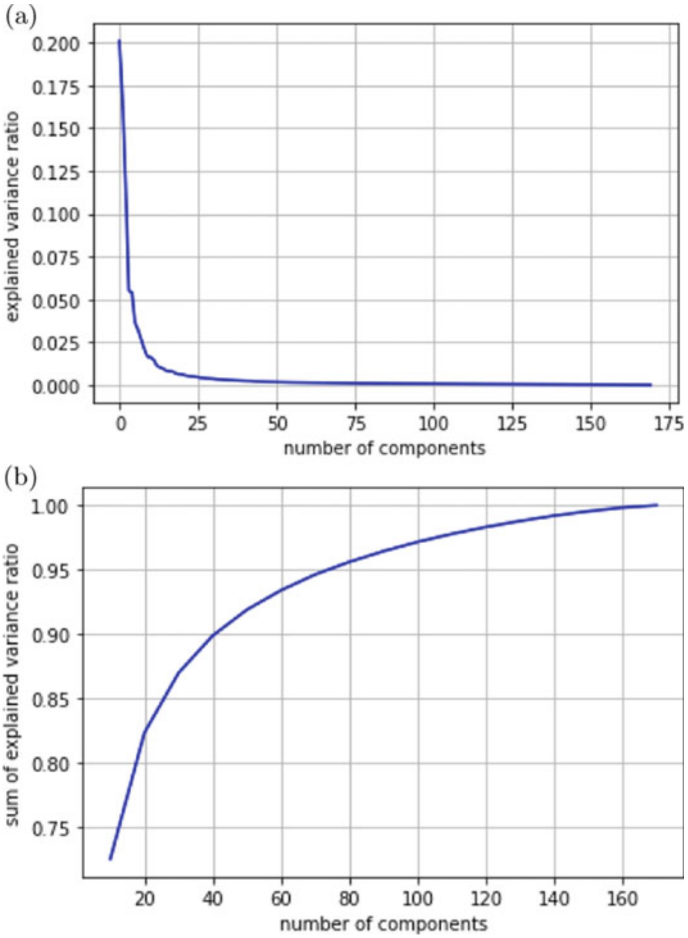
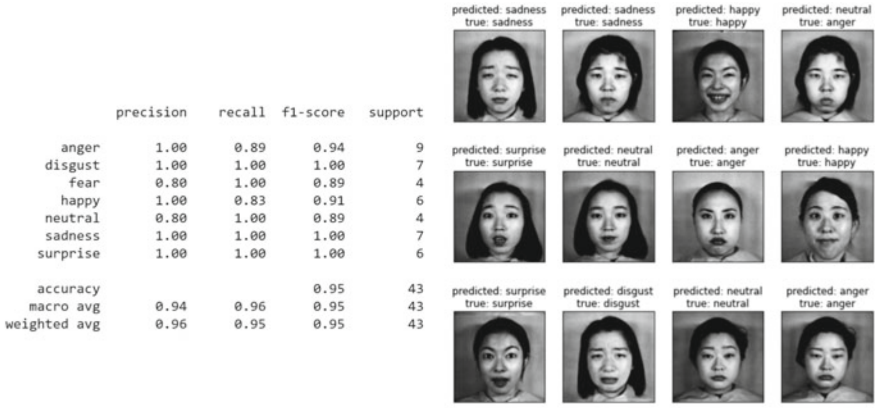


Fig. 12 **a** Explained variance ratio, **b** sum of explained variance ratio

have been proposed to extract features from the images, namely SNE [6], GPLVM [5], NMF [16] and LDA [13], after which SVM is used to classify the images. However, using PCA followed by SVM gives the best results with 95.35% accuracy. Abdulrahman and Eleyan [1] also used PCA followed by SVM; however, our model beats their performance. Our PCA + SVM model also beats the CNN model for the JAFFE data set [10, 11]. For the CK+ data set [9], the CNN model beats the PCA + SVM model marginally. Due the to PCA + SVM model being much less complex than the CNN model, we believe that the PCA + SVM model will be more efficient in real-time computing.



```
              precision    recall  f1-score   support

       anger       1.00      0.89      0.94         9
     disgust       1.00      1.00      1.00         7
        fear       0.80      1.00      0.89         4
       happy       1.00      0.83      0.91         6
     neutral       0.80      1.00      0.89         4
     sadness       1.00      1.00      1.00         7
    surprise       1.00      1.00      1.00         6

    accuracy                           0.95        43
   macro avg       0.94      0.96      0.95        43
weighted avg       0.96      0.95      0.95        43
```

predicted: sadness true: sadness
predicted: sadness true: sadness
predicted: happy true: happy
predicted: neutral true: anger
predicted: surprise true: surprise
predicted: neutral true: neutral
predicted: anger true: anger
predicted: happy true: happy
predicted: surprise true: surprise
predicted: disgust true: disgust
predicted: neutral true: neutral
predicted: anger true: anger

(a) (b)

Fig. 13 Results for PCA + SVM model for JAFFE data set [10, 11] **a** classification report, **b** sample predictions

Table 2 Proposed models versus other comparable models

Method	Data set	Accuracy (%)
SNE + SVM [6]	JAFFE	65.7
GPLVM + SVM [5]	JAFFE	65.24
NMF + SVM [16]	JAFFE	66.19
PCA + SVM [1]	JAFFE	87
LDA + SVM [13]	JAFFE	93.6
CNN [2]	JAFFE	86.38
CNN [14]	JAFFE	50.61
CNN [14]	CK+	65.54
PCA + SVM (proposed)	**JAFFE**	**95.35**
CNN (proposed)	JAFFE	87.50
PCA + SVM (proposed)	CK+	81.81
CNN (proposed)	CK+	83.84

6 Conclusions

This paper tried to address the facial emotion recognition problem using two popular methods—CNN and PCA with SVM. The proposed models were then tested on two different data sets—the CK+ data set and the JAFFE data set. The accuracy of the models is higher than the human level and can easily detect everyday emotions. With the increase in computer vision tasks, FER models will play an important role in evaluating user engagement for various commercial products. The source code of this work has been made publicly available at GitHub.[1]

Future work for this paper shall include employing these models for real-time emotion detection, which can be used by e-learning platforms to understand students' engagement. Another extension of the task attempted in this paper is to use multiple CNN networks instead of one and compare the accuracy of the same to a single CNN model.

References

1. Abdulrahman M, Eleyan A (2015) Facial expression recognition using support vector machines. In: 2015 23nd signal processing and communications applications conference (SIU), pp 276–279. https://doi.org/10.1109/SIU.2015.7129813
2. Dachapally PR (2017) Facial emotion detection using convolutional neural networks and representational autoencoder units
3. Ekman P, Friesen WV (1971) Constants across cultures in the face and emotion. J Personal Soc Psychol 17(2):124–129
4. Herba C, Phillips M (2004) Annotation: development of facial expression recognition from childhood to adolescence: behavioural and neurological perspectives. J Child Psychol Psychiatry Allied Disciplines 45:1185–1198. https://doi.org/10.1111/j.1469-7610.2004.00316.x
5. Huang MW, Wang ZW, Ying ZL (2010) A novel method of facial expression recognition based on GPLVM plus SVM. In: IEEE 10th international conference on signal processing proceedings, pp 916–919. https://doi.org/10.1109/ICOSP.2010.5655729
6. Huang M, Wang Z, Ying Z (2011) Facial expression recognition using stochastic neighbor embedding and svms. In: Proceedings 2011 international conference on system science and engineering, pp 671–674. https://doi.org/10.1109/ICSSE.2011.5961987
7. Krizhevsky A, Hinton G (2009) Learning multiple layers of features from tiny images. Techical report 0, University of Toronto, Toronto, Ontario
8. Liu K, Zhang M, Pan Z (2016) Facial expression recognition with cnn ensemble. In: 2016 international conference on cyberworlds (CW), pp 163–166. https://doi.org/10.1109/CW.2016.34
9. Lucey P, Cohn JF, Kanade T, Saragih J, Ambadar Z, Matthews I (2010) The extended Cohn-Kanade dataset (CK+): a complete dataset for action unit and emotion-specified expression. In: 2010 IEEE computer society conference on computer vision and pattern recognition—workshops, pp 94–101. https://doi.org/10.1109/CVPRW.2010.5543262
10. Lyons M, Kamachi M, Gyoba J (1998) The Japanese female facial expression (JAFFE) dataset, Apr 1998. https://doi.org/10.5281/zenodo.3451524
11. Lyons MJ, Kamachi M, Gyoba J (2020) Coding facial expressions with Gabor wavelets (IVC special issue). CoRR ArXiv:2009.05938, https://doi.org/10.48550/arXiv.2009.05938
12. Mehrabian A (1968) Some referents and measures of nonverbal behavior. Behavior Res Methods Instrument 1(6):203–207. https://doi.org/10.3758/BF03208096
13. Shah J, Sharif M, Yasmin M, Fernandes S (2017) Facial expressions classification and false label reduction using LDA and threefold SVM. Pattern Recogn Lett 139. https://doi.org/10.1016/j.patrec.2017.06.021
14. Shin M, Kim M, Kwon DS (2016) Baseline CNN structure analysis for facial expression recognition. In: 2016 25th IEEE international symposium on robot and human interactive communication (RO-MAN), pp 724–729. https://doi.org/10.1109/ROMAN.2016.7745199

[1] https://github.com/shivani-nandani/emotion-detection.git.

15. Yang D, Alsadoon A, Prasad P, Singh A, Elchouemi A (2018) An emotion recognition model based on facial recognition in virtual learning environment. Proc Comput Sci 125:2–10; The 6th international conference on smart computing and communications. https://doi.org/10.1016/j.procs.2017.12.003
16. Zilu Y, Guoyi Z (2009) Facial expression recognition based on NMF and SVM. In: 2009 international forum on information technology and applications, vol 3, pp 612–615. https://doi.org/10.1109/IFITA.2009.279

A Review on Deepfakes Detection Using Machine Learning Techniques

Abhinav Srivastava, Manish Pandey, and Santosh Kumar Sahu

Abstract Machine learning's recent developments have given rise to the phenomenon of "deep fakes," which is a cause for serious concern. Deepfakes are the phenomenon of creating fake digital items that look like real images, and a string of films has sprung up on social media in the past and coming years. Deepfakes are easily generated by anyone and can be easily distributed on social media owing to a low level of technical expertise. This paper aims to review various techniques that are used for the detection of deepfakes; many of the authors have used data sets of images and videos like FaceFroensics++, thispersondoesnotexist.com, etc. In various studies, some authors have used their own data for analysis. To accurately detect deepfakes, there is a need for smart technology. The study aims to show the existing technologies like convolutional neural networks, recurrent neural networks and support vector machines in the detection of deepfakes. In recent months, machine learning has gained popularity for detecting plausible face swaps in films that leave minimal indications of tampering or deepfake movies. Thus, to combat deepfakes, there is a need for efficient algorithms for detection in an early stage to stop blackmail, political unrest, etc.

Keywords Deepfakes · Artificial intelligence · Digital media · Digital image forensics

1 Introduction

Digital forensics is a branch of forensic science that deals with the recovery, analysis, and examination of digital devices and evidence in the context of criminal activities. For a few decades, the role of digital forensics was limited to combating cybercrimes. Digital gadgets, on the other hand, have become a source of evidence for crime detection as a result of technological advancement and its role in human life [1].

A. Srivastava (✉) · M. Pandey · S. K. Sahu
Department of Computer Science Engineering, Maulana Azad National Institute of Technology, Bhopal, India
e-mail: sriabhinav2@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_46

The field of image forensics research is a domain related to identifying fake images such as manipulation and tampering to stop the spread of fake content. Various approaches have been proposed for detecting forgeries in image, the majority of which either analyse inconsistencies relative to a camera pipeline or depend upon extracting image alterations in the resulting image which are specific. Image noise, for example, is an excellent indicator of splicing.

Deepfake is a method that seeks to manipulate the face of a famous individual in a video with the face of some other individual in order to spread fake news. It first came to the limelight in 2017 as a script for creating pornographic images or videos with switched faces. FakeApp has been developed by some online communities to enhance deep fakes.

The advancement in the fields of image processing, deep learning, and AI has given rise to deepfakes. A few years back, a video of the President of the United States circulated showing Barack Obama became viral after he was caught saying things he had never said. Deepfakes are harmful to society, humanity and are dangerous. In the twenty-first century with the advent of internet and social networking platforms such as Twitter, Facebook and WhatsApp, the spread of such content has become very easy, and if not stopped, it has the potential to exacerbate problems such as misinformation and conspiracy theories. In the beginning, big politicians, stars, actresses, and entertainers were the targets of deepfakes.

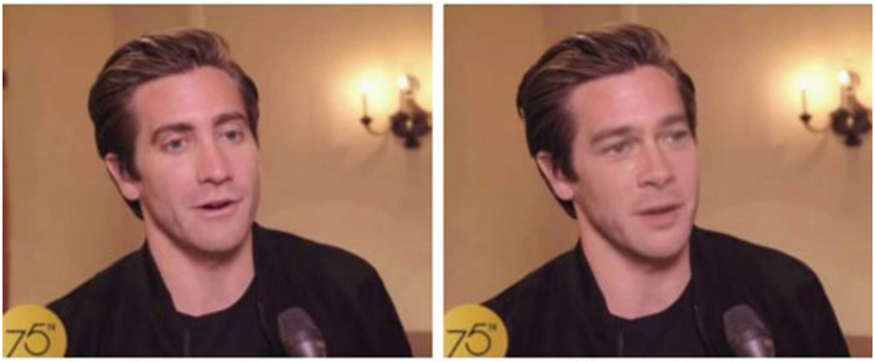
Videos of deepfakes look more realistic and are easier to create than classic videos of Hollywood, which are created by Abode, an image manipulation software. Face swapping is achieved in films by employing deep learning techniques with varied samples of video. The greater the samples are, the more realistic the results are.

The victims of deepfakes now include the general public, particularly women, and the majority of them are depicted as being in pornography. According to a post by Washington, pornographic images and face pics are being synthesised skilfully and are being circulated on social networks without seeking permission from third parties.

In recent years, deep learning techniques have proved to be quite successful in image forensics. Several researchers, such as Barni et al., employed deep learning algorithms in the detection of things locally in photos for JPEG compression. Bayar and Stamm [2] worked on generic image falsification. Deep learning algorithms seem to work efficiently in digital image forensics (Fig. 1).

2 Related Works

Deepfakes are becoming more dangerous to people's privacy, society's security and democracy. As the threat of deepfakes was surfaced, methods for identifying them were proposed. In the past, methods relied on features derived from flaws and glitches in the video synthesis process. In recent times, researchers have used deep learning techniques to extract prominent features automatically to detect deepfakes.



(a) Original image.

(b) Manipulated image.

Fig. 1 Deepfake manipulation [3]

Li and Lyu et al. [4] proposed an artificial intelligence (AI) method for detecting deepfake videos. The method proposed by them depends on an ML technique combating another ML technique. Convolutional neural networks (CNNs) were trained with real and manipulated figures. CNN networks were used for testing, with results ranging from 84 to 99% accurate. The results obtained by their technique achieved good accuracy.

Hu and Liao et al. [5] proposed a method of two stream by analysing the frame level and temporal level of compressed input video sequences. Redundancy to frames was added by compression. For extracting temporal correlation information, they used a temporal-level stream. To address the issue of deepfake films, temporal consistency was overlooked. In compressed deepfake video identification, the suggested solution outperforms state-of-art methods.

Han and Han et al. [6] suggested a two-stream network for detection of deepfake in video level, and the suggested approach is capable of dealing with low-resolution input. The suggested method dissects the video input sequence into segments before feeding selected frames from each segment into two streams: The first stream receives input from RGB data and analyses it to find out where it is contradicting semantically. Features of the noise recovered by spatially rich model filters were utilised by the second stream. They observed that conventional spatial-rich model filters with determined weights provide no substantial improvement, so they designed a learnable spatial-rich model filter that can better accommodate the noise inconsistency in tampering regions. In the final stage, stream fusion and segmental fusion are used to merge data from segments and streams. The method was tested on the largest FaceForensics++ and deepfake data sets.

Durall et al. [7] suggested a technique for detecting fake face images or deepfakes. Their technique utilises the analysis of frequency spectrum after which classifier is used. Concerning previous systems that required a vast amount of labelled data to function, the technique achieved great results with few training data and obtained great accuracy. The author created their own data sets by combining public data sets for obtaining high face resolution images such as Faces-HQ. The method achieved a classification accuracy of nearly 100 per cent after training on only about ten high-resolution images and around 90 per cent in the case of low-resolution images. The videos show their technique producing great results.

Jung et al. [8] proposed a technique for the identification of generative adversarial networks (GANs) generated deepfakes by DeepVision algorithm to analyse blinking pattern changes, which is a spontaneous action. The human eye pattern is believed to vary from person to person depending on the human physical state and various biological patterns. The pattern is influenced by the gender or age of the person, as well as the time of day, for example. As a result, deepfakes can be detected by using a pragmatic approach based on machine learning algorithms, to track changes in the blinking patterns of eyes that are significant.

Yang et al. [9] suggested a novel technique to reveal fake, tampered images or videos. The method is based on the discovery that deepfakes are created by splicing a synthesised region of interest in the face into the original image, resulting in imperfection that can be detected when three dimensional head postures are obtained from face images. Experiments have been conducted to demonstrate this phenomenon, and a categorization system based on this cue is being developed further. A collection of actual photos of faces and deepfakes is used to evaluate an SVM classifier, which employs features based on this cue.

Hashan and Salah et al. [10] introduced a procedure and expansive structure for following and following the provenance and history of computerised items back to their unique source utilising Ethereum keen agreements, regardless of whether the advanced thing is replicated on various occasions. The interplanetary document framework (IPFS) hashes are used in the keen agreement to store advanced data and metadata. Although the arrangement is centred around video content, the arrangement structure introduced in this exploration is sufficiently broad to be utilised for some other kind of advanced substance. Their technique depends on the possibility that if content can be truly followed to a respectable or confided in source, it can be trusted as real.

Guera et al. [11] presented a technique for identification deepfake videos automatically. Specific features were extracted from the frame by the technique which deploys a convolutional neural network (CNN). The features obtained are used for training a recurrent neural network (RNN) that is utilised for classification, whether or not a video has been manipulated. The strategy was tested on a vast amount of deepfake films gathered from different sources. Shruti Agarwal and Hany Farid et al. [12] described a method for detecting tampered videos by taking advantage of the fact that the dynamics of the shape of the mouth (visemes) are seldom not consistent with a spoken phoneme. Phoneme viseme mismatches can thus be used to detect temporally geographically and changes. The technique demonstrated was robust and efficient in detecting deepfakes.

Afchar et al. [13] presented a method to identify face manipulation in videos effectively, with a special emphasis on the following techniques used for generating forged videos which resemble real images like Face2Face and deepfake. Digital image forensics conventional methods are not suitable for movies due to the compression, which severely degrades the data. The study used a deep learning method to create two networks that focus on mesoscopic features of the image. They put such fast networks to the test on a data set that already exists as well as one that was created from videos. The tests show a high success rate in detection, with deepfakes detected more than 98 per cent of the time and faces detected more than 98 per cent of the time.

2.1 Machine Learning Algorithms

Artificial intelligence (AI) is a domain of computer science concerned with the creation of smart algorithms. Artificial intelligence (AI) is a group of algorithms that assist in the creation of techniques that help in classification, etc.

Machine learning is a subset of artificial intelligence based on the technique of developing a model to learn from past data, find similarities and make predictions for future data.

Machine learning is divided into various categories.

- Supervised Learning: the input data are given to the model along with the result in this type of learning, and both the input and output data are labelled.
- Unsupervised Learning: the model receives only raw data as input in this method of learning. The model is self-paced. There is no labelled data to be found.
- Reinforcement Learning: this type of learning focusses on taking a step towards maximising rewards. It is a reward-based learning system in which we are rewarded positively for tasks that produce the desired output and penalised negatively for any deviation from the expected outcome. It is used in robotics and industrial automation.

Deep Learning: a subfield of artificial intelligence, employs neural networks. Neurons are found in neural networks that work in the same way that the human brain does.

Neural networks are classified into two types (Fig. 2).

- Shallow neural network: a neural network with one or two hidden layers, as well as one input and one output layer.
- Deep neural network: a neural network with several hidden layers, the input and output layers of deep neural networks are the same.

Convolution neural networks, recurrent neural networks, and long short-term memory networks are examples of deep learning algorithms. Many of these are used in a variety of applications, such as CNN, are used for image processing. Artificial intelligence has a wide range of scope in different domains. Machine learning algorithms are being applied by researchers and forensic examiners in different fields to test models' efficiency and robustness. Correct application of machine learning

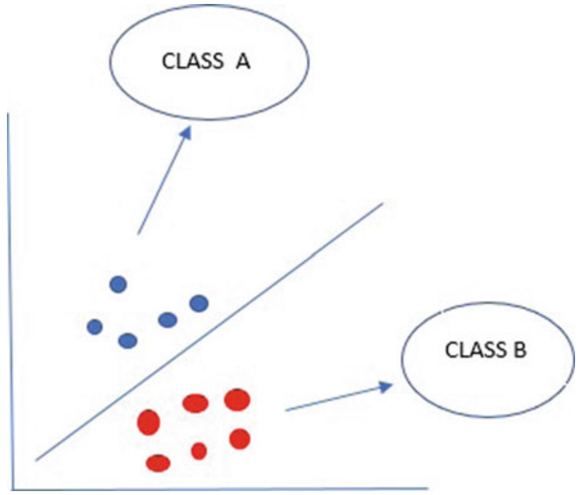


Fig. 2 Support vector hyperplane

algorithms is being utilised to produce effective results and thus help in identifying digital crimes.

2.2 Support Vector Machine

Support vector machine (SVM) is a machine learning algorithm that is supervised in its approach. SVM is a training model that uses training data for prediction. SVM is employed both for regression and classification problems, but mostly for classification problems, SVM is used for binary classification, and a hyperplane is found to separate the two classes. SVM uses a kernel trick to convert lower-dimensional data to higher-dimensional data [14].

2.3 Logistic Regression

The logistic regression algorithm is one of the basic classification algorithms. It is a statistical model that uses a logistic function. It is used for binary classification [15].

There are different types of logistic regression.

- Binary Logistic Regression: this type of regression has only two response, e.g. 0 or 1.
- Multinomial Logistic Regression: in this type of regression has three or more categories except for two, e.g. predicting the type of movies, e.g. action, thriller, SciFi.

- Ordinal Logistic Regression: in this type of regression, there are three or more categories.

2.4 K-Means Clustering

Clustering is a machine learning technique that is unsupervised and identifies similitudes in data points and clusters them. In clustering, first, some points, such as K, are initialised as means, and then, these points are assigned or categorised to their closest meaning, thus forming groups. The above process is repeated until all points are clustered [16].

2.5 Artificial Neural Network

Artificial neural network, or neural networks, have been widely used in deep learning. It functions similarly to human brains. The human brain consists of neurons that are million in numbers; there are chemical and electrical signals that pass from one neuron to the other neuron. Neurons are connected by structures called synapses [17].

- Input layer: is the first layer in a neural network architecture that receives external input.
- Hidden layer: is the middle layer in a neural network architecture; hidden layers can be one or more in number, and they are connected to the input.
- Output layer: is the final layer connected to the hidden layer; this layer includes an activation function.

2.6 Convolution Neural Network

CNN is a deep learning model that accepts an image as input and assigns significance to various objects in the image. ConvNet necessitates less pre-processing time than other techniques. CNN is an artificial neural network which is specifically used to process data which are images and is used in image recognition and processing. CNN has been applied in various domains, e.g. computer vision [18].

CNN has different architectures that are being used in deep learning (Fig. 3).

- Meso-4: this type of network has four layers of successive pooling and convolution. Then, there is a dense network that has one hidden layer. ReLU activation is used by convolutional layers, due to which nonlinearities are introduced, and batch normalisation is done to avoid vanishing gradients and regularise the output, [19] and dropout is performed in the layers [20].

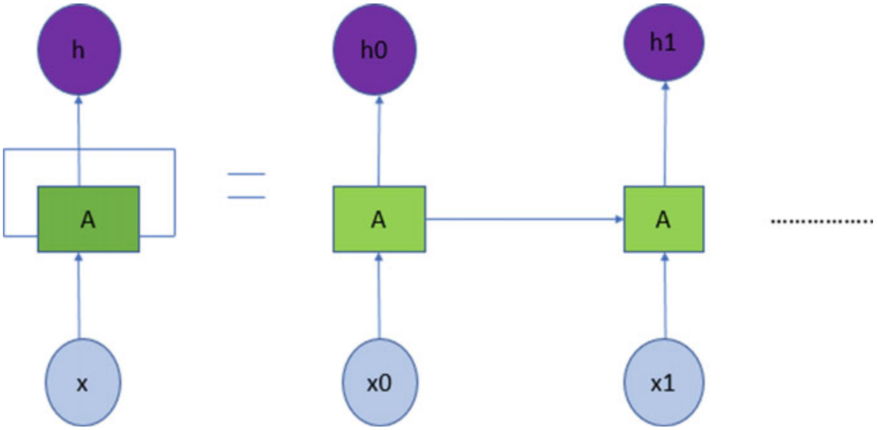


Fig. 3 Recurrent neural network

- MesoInception-4: another structure was introduced by Szegedy et al. by replacing the starting two layers of convolutional layers with the inception module [21]. Different kernel shapes are stacked with the output of various convolutional layers, thus optimising the functional space in the model. 3 × 3 dilated convolutions are proposed rather than conventional 5 × 5 convolutions. The proposal of using dilated convolutions with this module of inception as a way to deal with multi-scale input [22]. Replacing two layers offered better classification results rather than replacing more than two layers.

2.7 Recurrent Neural Networks

A recurrent neural network is a neural network used for processing a data sequence $x(t)$ with t varying from 1 to where t denotes the time. RNN is used for sequential input tasks involving language and speech. It is important to know the sequence of words to predict the next word in NLP. RNN outputs are dependent on the previous outputs, so RNN is called recurrent [23]. As a result, it can exhibit temporal dynamic behaviour.

RNN like any other architecture consists of

- Input: a time step $x(t)$ is taken as an input to the network.
- Hidden state: this layer act as memory for the network, $h(t)$ represents hidden state of the network.
- Weights: the inputs to RNN are parameterized by weight matrix.
- Output: the output of the network is denoted by $o(t)$.

2.8 Long Short-Term Memory Networks

- Long short-term memory networks are a form of recurrent neural networks that can learn from dependencies that are long term. Hochreiter and Schmidhuber discovered them, and many others refined them in subsequent research work, making them popular [24].
- These networks work splendidly on a broad range of tasks and are prominently used in the present scenario. LSTMs are specifically structured to circumvent the issue of long-term dependency. Their default behaviour is to remember things for a long period [25].
- RNN takes the form of neural network modules which are repeated in the chain. The standard RNN repeating modules are very simple containing a tanh layer.

2.9 Table of Comparison

Technique	Data set	Accuracy
CNN + LSTM [26]	Face2Face [27]	95
SVM [28]	Imagenet [29]	92
K-means [7]	CelebA, FlickrFaces-HQ	94
Logistic regression [7]	FlickrFaces-HQ	91
Meso-4 [13]	Face2Face [27]	89.1
MesoInception-4 [13]	Face2Face [27]	91.7

The above table shows various methods utilised for deepfake detection, so different techniques achieve different results. It must be noted that the data sets used by some authors are made on their own, and thus, accuracies can be different for different methods. The data sets like Face2Face, Faces-HQ and Imagenet. The data sets used are videos and images. Some of the data sets are images, and some as utilised by authors are videos. The accuracy corresponding to different techniques is different, and it can be seen in various studies that by increasing the feature extraction, the accuracy is increased. It can be seen that SVM is showing better classification accuracy than the other techniques on higher resolution images, whereas for lower resolution images, it shows classification accuracy of around ninety per cent.

3 Conclusion

The dangers of face alteration in the video are well-known these days. In real-world internet diffusion conditions. It is, nonetheless, critical to be able to comprehend the origins.

In this paper, various techniques were studied regarding the efficacy of methods to expose deepfake images. There are a lot of challenges in detecting deepfake images like these. Low-resolution images are harder to identify due to small frequency spectrum. Several machine learning approaches utilised power spectral features of the images to detect fake images and the algorithms like SVM, K-means and logistic regression achieved a decent classification accuracy. Nonetheless, a popular benchmark shows that several technologies can detect low-resolution forgeries with 91 per cent accuracy. The images that are compressed or resized have shown lower detection by the algorithms. Another challenge that the study found is that the majority of data sets available are inadequate, and thus, no public data set available is complete. Machine learning thus can be helpful in the detection of deepfake images and has produced good results and accuracy.

References

1. Iqbal S, Alharbi SA (2020) Advancing automation in digital forensic investigations using machine learning forensics. Dig Forensic Sci 3
2. Bayar B, Stamm MC (2016) A deep learning approach to universal image manipulation detection using a new convolutional layer. In: Proceedings of the 4th ACM workshop on information hiding and multimedia security, pp 5–10
3. Li Y, Yang X, Sun P, Qi H, Lyu S (2020) Celeb-df: a large-scale challenging dataset for deepfake forensics. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 3207–3216
4. Li Y, Lyu S (2018) Exposing deepfake videos by detecting face warping artifacts. arXiv preprint arXiv:1811.00656
5. Hu J, Liao X, Wang W, Qin Z (2021) Detecting compressed deepfake videos in social networks using frame-temporality two-stream convolutional network. IEEE Trans Circuits Syst Video Technol
6. Han B, Han X, Zhang H, Li J, Cao X (2021) Fighting fake news: two stream network for deepfake detection via learnable SRM. IEEE Trans Biometr Behav Identity Sci
7. Durall R, Keuper M, Pfreundt FJ, Keuper J (2019) Unmasking deepfakes with simple features. arXiv preprint arXiv:1911.00686
8. Jung T, Kim S, Kim K (2020) Deepvision: deepfakes detection using human eye blinking pattern. IEEE Access 8:83144–83154
9. Yang X, Li Y, Lyu S (2019) Exposing deep fakes using inconsistent head poses. In: ICASSP 2019–2019 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, pp 8261–8265
10. Hasan HR, Salah K (2019) Combating deepfake videos using blockchain and smart contracts. IEEE Access 7:41596–41606
11. Gu¨era D, Delp EJ (2018) Deepfake video detection using recurrent neural networks. In: 2018 15th IEEE international conference on advanced video and signal based surveillance (AVSS). IEEE, pp 1–6
12. Agarwal S, Farid H, Fried O, Agrawala M (2020) Detecting deep-fake videos from phoneme-viseme mismatches. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, pp 660–661
13. Afchar D, Nozick V, Yamagishi J, Echizen I (2018) Mesonet: a compact facial video forgery detection network. In: 2018 IEEE international workshop on information forensics and security (WIFS). IEEE, pp 1–7

14. Joachims T (1998) Making large-scale SVM learning practical. Technical report, Technical report
15. Wright RE (1995) Logistic regression
16. Likas A, Vlassis N, Verbeek JJ (2003) The global k-means clustering algorithm. Pattern Recogn 36(2):451–461
17. Sharma S (2017) Artificial neural network (ANN) in machine learning. Data Sci Centr
18. O'Shea K, Nash R (2015) An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458
19. Ioffe S, Szegedy C (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. In: International conference on machine learning. PMLR, pp 448–456
20. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 15(1):1929–1958
21. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: 2015 IEEE conference on computer vision and pattern recognition (CVPR). IEEE, pp 1–9
22. Shi W, Jiang F, Zhao D (2017) Single image super-resolution with dilated convolution based multi-scale information learning inception module. In: 2017 IEEE international conference on image processing (ICIP). IEEE, pp 977–981
23. Zaremba W, Sutskever I, Vinyals O (2014) Recurrent neural network regularization. arXiv preprint arXiv:1409.2329
24. Greff K, Srivastava RK, Koutnık J, Steunebrink BR, Schmidhuber J (2016) LSTM: a search space odyssey. IEEE Trans Neural Netw Learn Syst 28(10):2222–2232
25. https://colah.github.io/posts/2015-08-Understanding-LSTMs/
26. Yadav D, Salmani S (2019) Deepfake: a survey on facial forgery technique using generative adversarial network. In: 2019 international conference on intelligent computing and control systems (ICCS). IEEE, pp 852–857
27. Thies J, Zollhofer M, Stamminger M, Theobalt C, Nießner M (2016) Face2face: real-time face capture and reenactment of RGB videos. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2387–2395
28. McCloskey S, Albright M (2019) Detecting gan-generated imagery using saturation cues. In: 2019 IEEE international conference on image processing (ICIP). IEEE, pp. 4584–4588
29. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L (2009) Imagenet: a large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. IEEE, pp 248–255

Game of Definitions—Do the NIST Definitions of Cloud Service Models Need an Update? A Remark

Vishal Kaushik, Prajwal Bhardwaj, and Kaustubh Lohani

Abstract Cloud computing captured the world by storm as a cheaper, faster and safer alternative to an on-premises computing environment. Consequently, several organizations started favouring cloud computing service models over traditional on-premises hosting to offer their products or services over the Internet. As cloud computing gained popularity in the early 2000s, the National Institute of Standards and Technologies (NIST) decided to define it in 2011 formally. Since then, almost a decade has passed and cloud technology has progressed by leaps and bounds. As a result of the commercial progress that cloud computing has made as a technological offering, delivery methodologies have evolved and cloud service providers nowadays are using new service models that are not falling in the traditional service brackets of Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS) as defined by the NIST. Serverless and containers are examples of such service models. Consequently, cloud service providers developed novel terminologies like Database as a Service and Storage as a Service to clearly define their offerings. After researching and studying various examples and models, we feel the times have changed. Since the NIST definitions have not been updated since 2011, a lot of confusion has emerged among services that hinder compatibility. In this paper, we put forth the argument to update the definition of cloud service models introduced by the NIST in 2011. The argument is supported by detailing two new service methodologies: serverless and containers and explaining how these do not fit in the buckets defined by the NIST in 2011.

Keywords NIST definitions · Cloud computing · Serverless · Containers · XaaS · Container · Cloud service delivery models

V. Kaushik · P. Bhardwaj · K. Lohani (✉)
School of Computer Science, University of Petroleum and Energy Studies, Dehradun 248007, India
e-mail: kaustubhlohani25@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_47

1 Introduction

Cloud computing can be defined as a technology that enables sharing of configurable computing resources over a network that can be rapidly allocated with minimal client or vendor intervention [1]. There are several advantages to utilizing the cloud, such as low costs, better flexibility and better scalability; furthermore, it saves time required by IT personnel to set up traditional systems.

Since the last decade, cloud computing has become the backbone of modern technology. However, the cloud computing model has gone through many stages of evolution, starting from delivering applications and finally to worldwide delivery of computing resources through the Internet.

Cloud computing as a concept can be traced back to the 1960s with the introduction of ARPANET. ARPANET was the first time that the idea of globally interconnected computing devices was first conceived and led to grid computing development. The most significant change came in the 1990s when the Internet bandwidth became sufficient to boost web development. This boom fueled the rise of the Internet as we see it today. Soon, the first enterprise applications were served with the help of a web application by salesforce. Amazon launched Elastic Compute Cloud (EC2) and Simple Storage Service (S3) in 2006. This competition intensified in 2009 with the launch of browser-based applications like Google Apps and Microsoft Office Suite. Soon enough, in September of 2011, National Institute of Standards and Technology (NIST) decided to formally define the cloud paradigm and its service models, namely Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS).

Since then, a decade has passed and new service technologies like serverless and containers have come, allowing organizations to streamline their cloud implementation. Newer technologies use distinct delivery methodologies, resulting in them falling out of the traditional services buckets of IaaS, PaaS and SaaS. Considering the fast-paced manner in which cloud technologies are evolving, it might be wise to expand the scope of these buckets or redefine them altogether.

One of the popular approaches among current cloud service providers (CSP) and other organizations is the Anything-as-a-Service model, which no longer defines service in the traditional bucket of infrastructure, software or platform. It rejigs those buckets and redefines SaaS for SQL server as Database as a Service.

In this paper, we argue that NIST definitions of cloud service models are getting outdated and need to be updated or redefined altogether. In the subsequent sections, we will introduce the NIST service models and then define two new technologies (serverless and containers) that are not falling in the NIST-defined categories. We will introduce serverless computing and containers as a service model, compare them with the NIST categories of cloud service models and detail their differences effectively, concluding why they cannot be placed in the NIST-defined buckets of cloud service models. Finally, we would detail two novel frameworks, Anything as a Service (XaaS) and Johan Den Haan's framework, capable of potentially replacing the existing NIST service models.

2 Definitions

In this section, we introduce the NIST service models. This section will serve as a precursor to subsequent sections where we will present our arguments as to why these definitions should be updated or modified.

2.1 Infrastructure as a Service or IaaS

As defined by the NIST, IaaS is a cloud service model in which the CSP allocates computational resources to the consumer to enable application development and deployment. The applications that can be deployed over an IaaS can include an operating system and other custom software. IaaS products do not offer control of the underlying cloud infrastructure; however, the client controls deployed software, storage and select networking aspects [1]. Businesses commonly use IaaS to quickly set up and dismantle test and development environments, big data analysis, high-performance computing, storage backup and recovery and website hosting [2].

2.2 Platform as a Service or PaaS

As per the NIST, PaaS provides the consumer with the capability to develop or deploy applications on the cloud infrastructure where applications can be developed using tools, technologies and external libraries supported or provided by the CSP. PaaS service model does not allow the client to control the underlying cloud infrastructure (processing, networking and storage capacities and capabilities) and operating systems. However, the client has control over the application-hosting environment [1]. Businesses commonly use PaaS for the development framework and business intelligence [2].

2.3 Software as a Service or SaaS

As per NIST, SaaS is a cloud service model that allows consumers to utilize the applications developed by the CSPs and deployed over the cloud. The applications are made available to the client using an executable application package or through a supporting interface such as a web browser. SaaS does not offer the clients the freedom to customize the application capabilities, with the possible case of managing client-specific application configuration settings such as user IDs and passwords. Moreover, the management of underlying cloud infrastructure, including operating system, processing, networking and storage capacities and capabilities, also rests

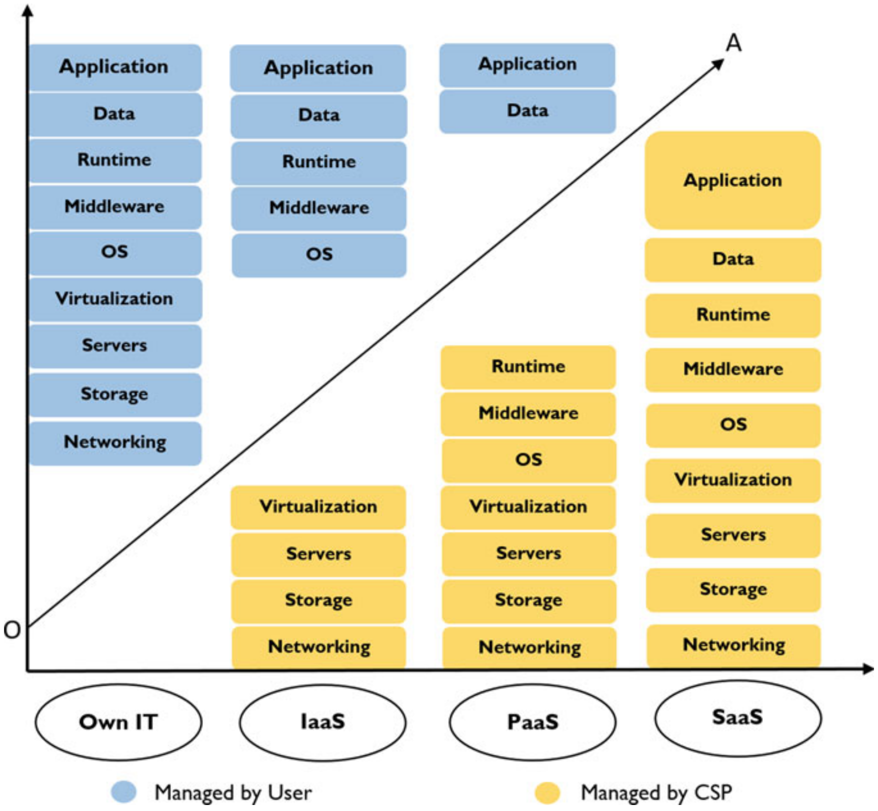


Fig. 1 Pictorial representation detailing the differences between Own-IT versus IaaS, PaaS and SaaS (every layer above line OA is user-managed, and every layer below OA is vendor managed). Here, all the layers are depicted from the perspective of the CSP. However, in a cloud offering, the vendor provides control of these layers to the user

with CSP [1]. Common examples of SaaS used are web-based email services such as Gmail, Netflix and MS Office Suite. Businesses use SaaS as customer relationship management (CRM) and enterprise resource planning (ERP) tools [2] (Fig. 1).

3 Serverless Computing

Serverless is a cloud service methodology where the client only manages the hosting configurations and environment settings. At the same time, the CSP allocates the server based on the clients' desired specifications and further takes over the management of code executions over those servers [3].

Serverless is one of the recent innovations that are not reasonably placed anywhere in the bracket of traditional service models like IaaS, PaaS and SaaS. Serverless computing uses a pay-as-you-go subscription model to provide application development and deployment environment to the consumers. Apart from an attractive subscription model, the primary appeal for clients towards serverless is the ability to develop and deploy applications without the burden of managing the underlying servers or code execution setup. An organization that chooses a serverless vendor over a conventional one does not need to reserve bandwidth and pay for the whole; instead, they pay for the computational resources they utilize as the service is auto-scaling [4, 5].

Serverless computing is an escape from costly and effort-consuming infrastructure (servers) management required to launch a product. Indeed, serverless as a term can be misleading as physical servers are still involved. However, CSPs take on the burden of managing the physical infrastructure rather than the client, making the service model "server less" from the clients' point of view.

Serverless computing provides a unique amalgamation of features from each of the traditional service models. Consequently, it is not entirely accurate to categorize it like IaaS, PaaS and SaaS.

Serverless hosting providers: Lambda (AWS), Azure Functions, Cloud Functions (Google Cloud), OpenWhisk (IBM Cloud), Oracle Functions (Oracle) and Cloudflare Workers (Cloudflare).

3.1 Comparison Between Serverless and PaaS

Both PaaS and serverless by design keep the entire back-end architecture invisible to the developers and let them exercise a similar level of control; one might argue that both are similar. This is far from being a complete argument.

The NIST definition of the PaaS service model mentions that the client does not control the underlying cloud infrastructure (processing, networking and storage capacities and capabilities) and operating systems. However, the client has control over the application-hosting environment [1]. According to this, the PaaS users are expected to manage application deployment and hosting environment configurations. In contrast, serverless provides lesser user control over hosting to provide a more straightforward deployment, easy scaling and a pay-as-you-go subscription model. In the serverless model, the CSP runs the user-provided code whenever prompted while maintaining the hosting environment variables.

There are two crucial differences between PaaS and serverless: scalability and subscription model. In PaaS, the consumer pays for a whole platform; in contrast, serverless offers the client an opportunity to pay only for the resources utilized for the time utilized. Moreover, scalability in the two models works differently. Serverless provides auto-scaling, enabling the user to scale automatically when required, unlike PaaS, where the user first needs to think about structuring and scaling the platform [6].

One can compare serverless computing to oil coming from a pipeline. Oil from a pipeline is unlimited as long as the connection is intact, meaning that amount of oil available for use at a given time can be increased or decreased as per the change in requirement in real time. PaaS, however, is much similar to oil delivered through ships in a limited quantity at a time, meaning the supply can be potentially unlimited, but there is only limited oil available for use at a given point in time. However, the client can get more oil on demand, but it is not that simple; the client needs to get in touch with the supplier nation to deliver more if the requirement arises. In both cases (pipeline and ship delivery), an external vendor handles the back end—oil procurement, filtration and delivery. However, only oil from the pipeline can be scaled precisely in a time-bound manner. Moreover, procuring oil from the shipping route could mean that the client would need to pay for it even if they cannot utilize it whereas, one can get the precise amount required from the pipeline, potentially saving money. Thus, serverless has significant advantages over PaaS in the areas of scaling, flexibility and cost.

In Fig. 2, we have demonstrated that the user has control over the data, application back end and application client-side in the PaaS service model. Essentially, the application and all the data are controlled by the user. In contrast, in the serverless model, the user has partial control over the application back-end and complete control over the client-side of the application. Moreover, in serverless, the user has no control over the platform data or configuration. In the serverless model, the user uploads the application code and sets the manner of execution. The CSP takes care of code execution in a user-specified manner and presents the user with the output.

Finally, we can conclude that there are significant differences between PaaS and serverless cloud service models.

3.2 Comparison Between Serverless and SaaS

As defined above, SaaS is a software distribution model that provides the consumer with a finished ready to use software product that requires no development input in the form of code, libraries. Serverless and SaaS have similarities: they have no client-side hardware requirement for deployment, no server processes to manage, inherent scalability and high availability.

According to the NIST definition of SaaS, it does not offer a client the freedom to customize the application capabilities, with the possible case of managing client-specific application configuration settings. Moreover, the client is not given the management of underlying cloud infrastructure, including operating system, processing, networking and storage capacities and capabilities [1]. So, as per NIST, SaaS offers the least amount of infrastructure and environmental control to the user. On the other hand, serverless offers some control of the platform. Scaling is another area where serverless is more flexible as it offers auto-scaling.

On the contrary, scaling up in SaaS is not transparent; it depends on the pricing and quality. Charges for SaaS services are on a subscription-based model; users are often incentivized to take up a more extended subscription and pay for possibly more than they can consume, effectively taking away their ability to scale up and down as

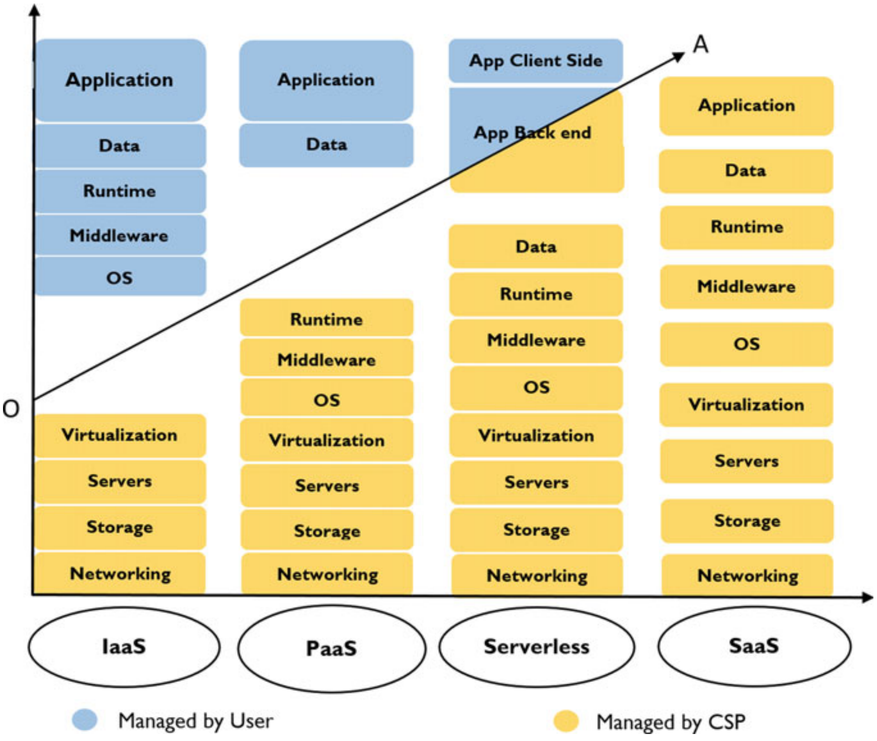


Fig. 2 Pictorial representation detailing the level of control offered in traditional service models and serverless (here, every layer above line OA is user-managed and every layer below OA is vendor managed). Here, all the layers are depicted from the perspective of the CSP. However, in a cloud offering, the vendor provides control of these layers to the user

they need. On the other hand, serverless computing provides auto-scaling and pay-as-you-go facilities; these features make serverless computing different from SaaS offerings.

In Fig. 2, we have shown the difference in the level of control offered to the user in serverless and SaaS service models. As depicted, SaaS offers no user control over the layers, whereas, in the serverless model, the user has partial control over the application back-end and complete control over the application client-side.

Thus, we can conclude there are significant differences in the SaaS and serverless cloud service models.

3.3 Comparison Between Serverless and IaaS

IaaS is a cloud delivery methodology that gives the consumer complete control over the server infrastructure. Organizations opting for IaaS get the freedom to set up storage, networks and other computational resources as per their requirements. Serverless computing takes away most of the control in exchange for more straightforward implementation without worrying about infrastructure issues such as allocation of VM resources, server availability and management and server scaling.

Furthermore, serverless is more easily scalable than IaaS and features the pay-as-you-go-model, letting users pay only for resources they are utilizing. These differences between IaaS and serverless are evidence that serverless cannot be classified as an IaaS service model.

Moreover, as depicted in Fig. 2, IaaS offers operating system, middleware, runtime, data, application client-side and application back-end control to the user. In contrast, serverless offers partial application back-end and complete application client-side control to the user. Thus, solely based on the level of control offered to the user, serverless cannot be classified the same as an IaaS service model.

4 Containers

Containers are executable units of software in which all essential libraries and external dependencies that support application development and deployment are packaged such that it can run anywhere in a traditional IT or a cloud environment [7, 8].

Traditional VMs virtualize the underlying hardware and divide it so that multiple OS can be installed upon which application code and libraries are placed to run. In contrast, containers virtualize the OS placed upon hardware by using a container engine such as Docker. The absence of guest OS is the reason why containers are so lightweight, thus fast and portable (Fig. 3).

Container service providers. Google Container Engine (GKE), Amazon EC2 Container Service (ECS) and Azure Container Service (ACS).

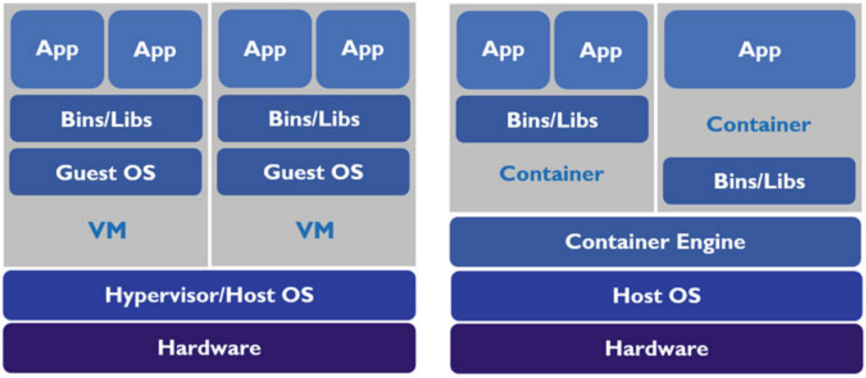


Fig. 3 VM architecture versus container architecture

Since containers are virtualized over the guest OS instead of hardware level, the resulting model has many advantages over other traditional cloud service models. Containers can be used by an organization that wants infrastructure control of IaaS and the flexibility of PaaS [9].

Some of the critical features of containers are:

- *Portability*: Having the application and all of its dependencies in a single container provides the user with the unique feasibility of running the application in various environments, public and private clouds, increasing flexibility and reducing the implementation time.
- *Highly efficient and cost-effective*: A container does not require a guest OS to operate, which reduces the computational power needed to run a single container. Consequently, a single server can be used to run several containers, which otherwise would be used to run VMs. This efficiency, coupled with a higher hardware utilization percentage, helps reduce data centre and bare metal costs.
- *Security*: Isolation among containers serves as a security feature by hindering any inter-container malware infection.
- *Speed*: Since container implementation does not require setting up a guest OS from scratch, it takes seconds to generate, dismantle, or mirror, containers enabling a quick development process and faster product/service launch speeds.
- *Scaling*: Containers have a unique feature of horizontal scaling, allowing users to increase the number of containers on a single cluster, thus enabling the organization to scale out as per requirements.

4.1 Comparison Between Containers and PaaS

PaaS is a developer-focused solution for application delivery and deployment that integrates necessary tools and technologies in a single offering. In contrast, container services are different; they provide a complete set of tools for creating and managing containers. Essentially, PaaS is a platform that facilitates application development and deployment using a single product. In contrast, containers require the user to deploy all the application code and dependencies as per the requirements on these containers. Essentially, containers offer users a higher level of environment control as compared with PaaS.

As depicted in Fig. 4, containers offer users partial control over the operating system kernel absent in the PaaS service model. Furthermore, users can manage middleware and configure runtime in the container offerings. In contrast, middleware and runtime configurations are taken care of by the CSP in the PaaS service model.

Deciding whether an organization can benefit from using PaaS is a different debate than whether they should use containers or not. In general, developers appreciate the flexibility of containers but prefer the control of the PaaS model.

Container solutions can integrate with PaaS in some scenarios. For example, AWS Beanstock (PaaS) is used to develop applications and then deploy on ECS (container offering); this is an example of a PaaS-container hybrid.

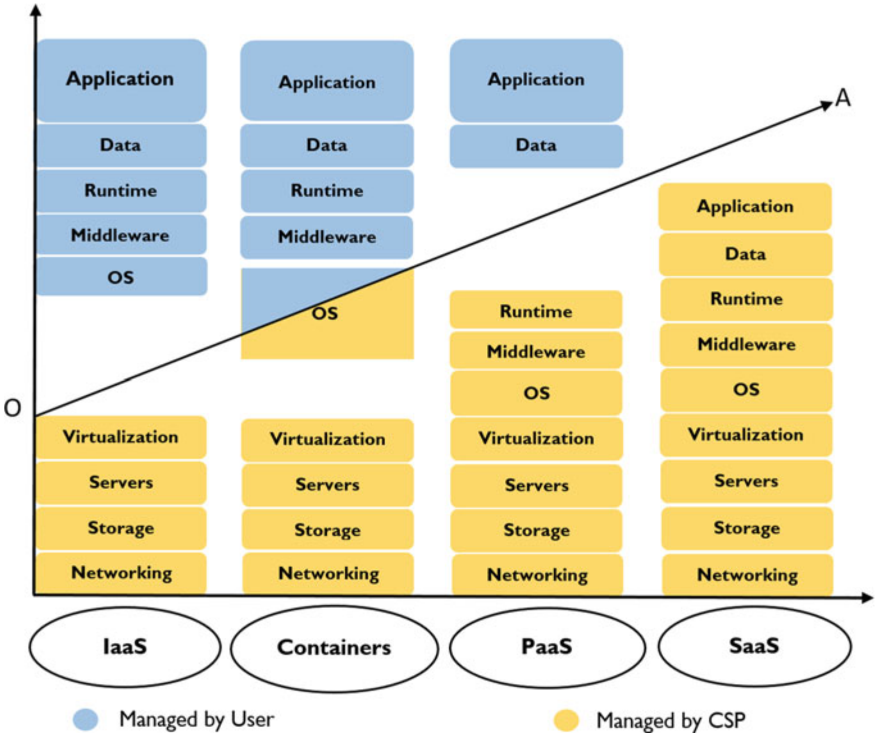


Fig. 4 Pictorial representation detailing the level of control offered in traditional service models and containers (here, every layer above line OA is user-managed and every layer below OA is vendor managed)

4.2 Comparison Between Container and IaaS

Considering the above-listed features of containers, one can argue that it falls in between IaaS and PaaS. The current container service model provides hardware-level control to the consumer by using containers as its basic resource unit, unlike IaaS, which uses VMs (as evidenced from Fig. 3). Container virtualizes over an IaaS service. In contrast, IaaS runs on data centres. Typically, an IaaS provider will primarily deliver on-demand VMs. In contrast, a container provider will provide managed service of containers, making horizontal scaling easier and enhancing security and application portability.

Moreover, as depicted in Fig. 4, IaaS gives the user complete control of the operating system kernel, meaning the user can effectively change the operating system without the involvement of the CSP. However, this is not the case with the container, where the user has partial control over the deployed operating system.

Thus, classifying the container service model same as IaaS would not be entirely accurate.

4.3 Comparison Between Container and SaaS

SaaS is a complete end-to-end product offering that the customers choose because of the software functionality. On the other hand, containers use a virtualization methodology that can deploy a custom-made application. So essentially, SaaS is an entirely software-based solution that serves a particular job. The container service model uses virtualization and creates several container instances on which an application code can be hosted with all the dependencies.

Furthermore, as shown in Fig. 4, containers provide significantly more control to the user than SaaS. Thus, classifying containers as the SaaS service model is not entirely accurate.

5 Final Thoughts

5.1 Serverless Computing

As described in Sect. 3, all the major CSPs have a serverless offering, proving that serverless is a popular cloud offering among consumers. However, as detailed in Sects. 3.2 through 3.4, incorporating serverless in the NIST-defined buckets of IaaS, PaaS and SaaS is not feasible because they have significant differences with the current serverless offering. Serverless as a service model falls right in between SaaS and PaaS, whereas, when compared with IaaS, serverless offers less control over the computational resources.

5.2 Containers

As evidenced from Sect. 4, all popular CSPs provide container service, proving that container as a service model is popular among organizations. Furthermore, as detailed in Sects. 4.3 through 4.5, the container service model cannot be accurately placed in any of the service models defined by the NIST.

5.3 Ambiguity in Offered Services

Not only serverless and containers but other services provided by the CSPs also generate confusion. Sometimes, a single service is classified as IaaS by one group, PaaS or SaaS by another group.

6 Way Forward

As many organizations adopt cloud technology, CSPs have introduced unconventional technologies like serverless computing and containers. Consequently, cloud services methodologies are moving away from the traditional service models: IaaS, PaaS and SaaS. Moreover, NIST has not updated these service models for a decade, resulting in new frameworks to clearly define the products offered under a broad umbrella of cloud computing.

Some of the popular frameworks capable of segregating the cloud delivery methodologies are listed below.

6.1 Anything as a Service (XaaS)

XaaS refers to a service previously provided on-site but now delivered over the Internet [11]. Advancements in the Internet have enabled organizations to present almost all of the IT services over it.

Recently, CSPs are taking up the approach of describing their service along the lines of XaaS as it allows them to respond quickly to market changes, thus becoming more flexible. Furthermore, one can argue that Database as a Service sounds clearer from the consumer's perspective than PaaS for the database. Moreover, a database can be offered through two NIST service models (IaaS and PaaS). So, naming the service as Database as a Service removes the confusion and focuses on the offered functionality.

Some examples for XaaS include:

Blockchain as a Service (BaaS): Provides consumers cloud-based services to create and manage blockchain applications. *Providers*: Azure Blockchain Service, Amazon Managed Blockchain and IBM Blockchain Platform.

Database as a Service (DBaaS): A cloud-based service provides consumers with a platform to manage their database via their preferred database technology without setting up physical servers and installing software. *Providers*: IBM Cloud Databases, Amazon RDS and Azure SQL Database.

Storage as a Service (STaaS): STaaS is a service model enabling renting physical storage usable over the Internet from a cloud vendor. The CSP also provides basic ways to access that storage. *Providers*: AT&T Synaptic Storage as a Service.

Disaster Recovery as a Service (DRaaS): Cloud-based service that offers business continuity solutions by restoring data, servers in the event of system failure. *Providers*: VMware Business Continuity and Disaster Recovery.

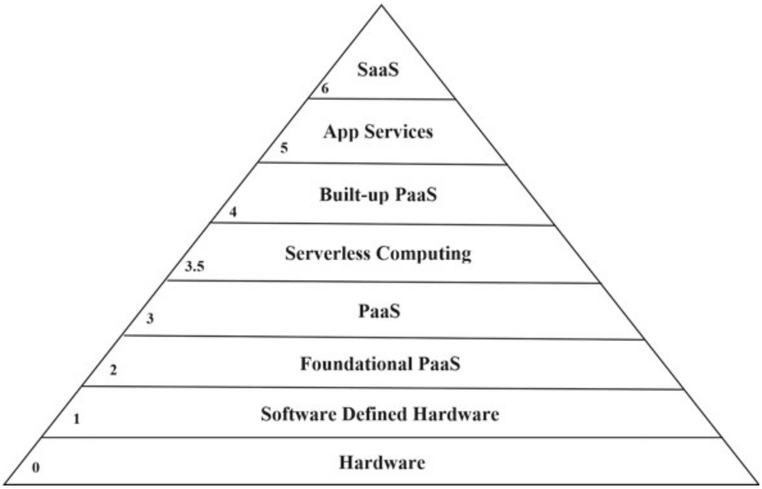


Fig. 5 Updated cloud service model suggested by Johan den Hann and further, modified by Christine Miyachi [13]

6.2 A New Model

Another approach for categorizing service models was given by Johan den Haan [12], CTO at Medix and further modified by Christine Miyachi [13] to include an additional layer for serverless computing. The model builds on the pre-existing NIST model and related ones, which are then integrated to form a new framework. Johan further divides PaaS into sub-layers to align the model to current commercial offerings (Fig. 5).

The proposed model helps organize current commercial cloud service offerings. The updated model also incorporates the latest offerings like serverless and container models.

The lines between the traditional NIST models are blurring, and IaaS, PaaS and SaaS cannot keep up, giving birth to numerous sub-categories describing many novel approaches.

DevOps can be categorized as Layer 2 service; Layer 3 is designed to accommodate products/services for a professional developer. Layer 4 services target the rapid developers—those who want to develop an application but are not well-versed in writing it from scratch. Layers 5 and 6 are a software offering suited for any and every user.

This model clearly shows that NIST definitions of IaaS, PaaS and SaaS are not holding up in place after technological advancements.

7 Conclusion

In this paper, we argued why the NIST definition of cloud service models needs an update. Moreover, we described the newer cloud service models such as serverless computing and containers. Furthermore, we supported our initial argument by explaining how the new service models are different and thus cannot be placed definitively in the predefined buckets of NIST-defined cloud service models (IaaS, PaaS and SaaS). Moreover, we mentioned two frameworks that can be potentially be used to replace or update the existing NIST definitions.

Finally, we would conclude by mentioning that as technology matures, the formal definitions should be updated to ensure they align with the latest technological advancements. Thus, after a decade has passed, we believe that perhaps the time has come to update NIST definitions for cloud service models and include more current industry-oriented technologies. For the time being, NIST definitions do well in broadly categorizing the cloud services and serve as a good platform for beginners to start in the cloud vertical. However, if no update is done, the NIST definitions might drift apart from the latest technological offerings as newer cloud delivery methodologies get introduced.

References

1. Mell P, Grance T (2011) The NIST definition of cloud computing
2. Buyya R, Broberg J, Gościński A (2011) Cloud computing: principles and paradigms. Wiley
3. Serverless computing. Wikipedia. https://en.wikipedia.org/wiki/Serverless_computing. Accessed 10 Aug 2021
4. McGrath G, Brenner PR (2017) Serverless computing: design, implementation and performance. In: 2017 IEEE 37th international conference on distributed computing systems workshops (ICDCSW). IEEE, pp 405–410
5. Van Eyk E, Toader L, Talluri S, Versluis L, Uță A, Iosup A (2018) Serverless is more: from paas to present cloud computing. IEEE Internet Comput 22(5):8–17
6. Castro P, Ishakian V, Muthusamy V, Slominski A (2019) The rise of serverless computing. Commun ACM 62(12):44–54
7. Erl T, Mahmood Z, Puttini R (2014) Cloud computing: concepts, technology and architecture. Prentice Hall
8. Pahl C (2015) Containerization and the paas cloud. IEEE Cloud Comput 2(3):24–31
9. Pahl C, Brogi A, Soldani J, Jamshidi P (2017) Cloud container technologies: a state-of-the-art review. IEEE Trans Cloud Comput 7(3):677–692
10. Sosinsky B (2011) Cloud computing bible. Wiley Publication
11. Gibson J, Rondeau R, Eveleigh D, Tan Q (2012) Benefits and challenges of three cloud computing service models. In: 2012 4th international conference on computational aspects of social networks (CASoN). IEEE, pp 198–205
12. Haan JD (2013) The cloud landscape described, categorized and compared. The Enterprise Architect. www.theenterprisearchitect.eu/blog/2013/10/12/the-cloud-landscape-described-categorized-and-compared
13. Miyachi C (2018) What is "cloud"? It is time to update the NIST definition? IEEE Cloud Comput 5(03):6–11

Phishing Site Detection Using Artificial Intelligence

Ameya Chawla and Sachwin Singh Kohli

Abstract Cyber security is becoming a crucial and indispensable component of the modern era wherein it can be estimated that there will be over 800 million utilizers of web services by the end of the year 2022. Therefore, a need for security solutions to safeguard the general population from phishing scams is of paramount importance as it not only has repercussions on monetary components but also has dire consequences on the psychological well-being of the general population, making them fearful of exploring and utilizing the World Wide Web which is the sole cause to get rid of the probable downsides by formulating reliably methodical frameworks. The sole intent of this endeavour is to target and scrutinize the recurrent idiosyncrasy displayed by several phishing web pages and develop a framework to ascertain such sites on the Internet. The proposed approach to solve the above-mentioned problem is the ensemble model of decision trees having max depth 18, and this ensemble model is created by using K-Fold cross-validation techniques where K models were created, and the max vote classifier made by an ensemble of all of them. 10 models were created, and accuracy for each model ranged from 93.12% to 98.28%. The above method was researched, and it was derived that it is implementable by deploying a web application wherein the utilizer can type the site URL and using the same, the end project would be able to generate figures for several criteria basis of which the developed model had been trained, which would be able to give an analysis to ascertain whether or not a website is fraudulent(Phishing) or legitimate.

Keywords Cybersecurity · Phishing · Decision tree · Max vote classifier · K-fold cross-validation

A. Chawla (✉) · S. S. Kohli
Guru Tegh Bahadur Institute of Technology, Guru Gobind Singh Indraprastha University,
New Delhi, India
e-mail: ameya.chawla.ml@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_48

1 Introduction

The term phishing in this context is used to signify criminal activity related to cybersecurity or, in simple terms, cybercrimes, wherein a hacker/technocrat transmits faux website URLs to a targets' console, either to gain confidential data from the users like their email, passkeys, government identification credentials, credit/debit card details, bank account information and so on. Such malicious sites can potentially lead to the event of downloading and onboarding of detrimental programmes via which the said hacker may attain complete authority over the users' console remotely. Most scam websites look evidently like the original website, which is being imposters, and when their domains are tallied with the legitimate site, there are very minute differences between the two, for instance, a typographical error or resembling characters as in the original to discombobulate the end user, leading the user to mistrust and give out sensitive data. The objective of this paper is to create a deployable solution to detect phishing websites efficiently, and to solve this problem, the ensemble model in this paper is proposed.

2 Literature Review

2.1 Phishing Website Detection Using Machine Learning Algorithms

Authors of this paper compared different machine learning algorithms like decision tree, support vector machine and random forest with other test and split while training and achieved highest accuracy of 97.14 on 9:1 train-test split with random forest model [1].

2.2 Phishing Website Detection Using Machine Learning Classifiers Optimized by Feature Selection

Authors of this paper proposed methodologies to select features for machine learning model and proposed a random forest model which achieved an accuracy of 100% [2].

2.3 Detection of Phishing Websites from URLs by Using Classification Techniques on WEKA

Authors of this paper compared different machine learning algorithms result with metrics accuracy, precision, recall and F_1 score. Algorithms used for comparison were random forest, logistic regression, Naïve Bayes and decision tree. Highest scores were obtained with random forest model with 83% as highest accuracy [3].

2.4 Detection and Classification of Phishing Websites

Authors of this paper compared different machine learning algorithms result with metric accuracy. Algorithms used for comparison were logistic regression, support vector machine, multilayer perceptron, autoencoder and XGBoost, where multilayer perceptron achieved highest accuracy of 85.8% [5].

2.5 Hybrid Rule-Based Solution for Phishing URL Detection Using Convolutional Neural Network

Authors of this paper compared many different machine learning and deep learning algorithms to detect phishing websites like decision trees, support vector machine, multilayer perceptron and convolution neural network. Highest accuracy was obtained on CNN of 97.945% [5].

3 Data Set

The phishing(fraudulent) website data set shared by the UCI in their machine learning repository has been referred for this project. The data set comprises data of over 11,000 websites with around 30 set parameters revolving around each website. All of the parameters have a value of either −1, 0, 1, based on which we formulate an integer value of −1, 1, which points out whether or not is the website tested fraudulent(Phishing). The data set is divided into a ratio of 9:1, where 90% of the sites are worked on for training our model and the remaining 10% of it were used for testing the trained model.

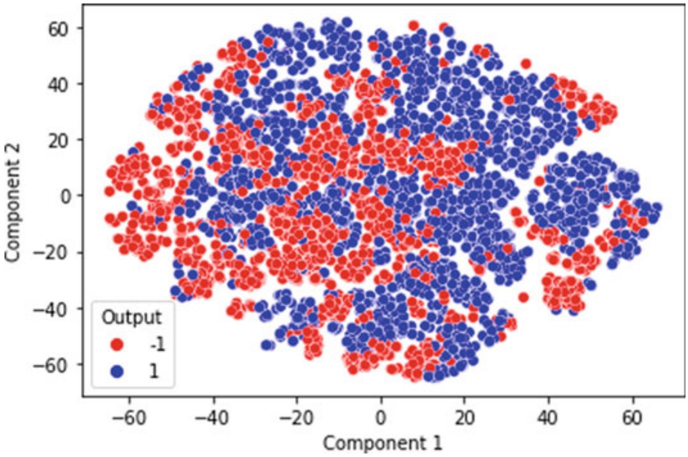


Fig. 1 t-SNE plot of two components after dimensional reduction

3.1 Data Visualization

Data set, which is used comprises 30 features, which in turn makes it a very highly dimensional data set and standard methodologies cannot be referred to plot the said data set, therefore, t-SNE is worked with to cause dimensional reduction by marginalizing the 30 parameters in the data into just 2 parameters, for the visualization and the exploratory data analysis of the data set we have used (see Fig. 1).

The plot in Fig. 1 indicates the separation between each action class and a nonlinear function can easily fit on this data set to solve the classification problem. It can be observed from Fig. 1 that there is overlap between clusters of both classes and any nonlinear complex function is perfect to solve this problem as it will be able to separate these points.

3.2 Data Cleaning

The given data set we have referred to does not hold any empty values or outlier values, hence, there are no modifications made in our data set.

3.3 Feature Selection

The wrapper-based feature selection is utilized to check out all of the plausible combinations and take into notice as to when the highest accuracy is reached, all the 30 parameters are taken into consideration.

3.4 Data Transformation

The data transformation was not utilized as the said data is already explained in the 3-integer set $\{-1, 0, 1\}$.

4 Data Set Parameters

The data set has over 30 parameters which were based on address, abnormal behaviour, HTML, JavaScript and domain details.

5 Validation Technique

Stratified K-Fold Cross-Validation technique is used where first the data set is randomly shuffled so that there should be no majority of a particular class in the testing data set. After mixing, the data set is divided into K groups of equal size, and the model is then trained K times, where each time model is training for data excluding those K samples, and after preparing the model, it is tested against those K samples and accuracy, precision, recall and $F1$ score is calculated.

Accuracy

$$\frac{\text{TrueNegative} + \text{TruePositive}}{\text{TrueNegative} + \text{TruePositive} + \text{FalsePositive} + \text{FalseNegative}} \tag{1}$$

It tells how many websites were classified correctly. It is a primary measure of performance of the model and higher this metric higher websites are ranked correctly.

Precision

$$\frac{\text{TruePositive}}{\text{TruePositive} + \text{FalsePositive}} \tag{2}$$

It tells how many predicted genuine websites were actually genuine. It is a comparison considering expected output and testing for predicted positives which are truly positive as an error in this metric are phishing websites classified as genuine websites.

Recall/Sensitivity

$$\frac{\text{TruePositive}}{\text{TruePositive} + \text{FalseNegative}} \tag{3}$$

It tells how many genuine websites were predicted correctly out of total real websites. It is comparison considering on expected output and testing for predicted positives which are truly positive as error in this metric are genuine websites classified as phishing websites.

F_1 Score

$$\frac{2 * \text{Recall} * \text{Precision}}{\text{Recall} + \text{Precision}} \tag{4}$$

F_1 score is used in the cases to judge when one model has better score in either recall and lacks in precision or vice versa.

6 Proposed Approach

Decision Tree is used in proposed approach to create models to train over the data set, where stratified *K*-fold cross-validation was used and *K* models were developed and final ensemble model of these *K* models having max vote as final layer was selected. Maximum depth used by the decision tree was 18, and ensemble model of 10 decision tree was created as value chosen for *K* in *K*-fold cross-validation was 10.

Algorithm defined to create the ensemble model [6, 7].

Algorithm 1: Algorithm to construct ensemble model using K-fold cross-validation

```
Input: Dataset
Output: Ensemble model and accuracies
K = 10 // Number of Folds
$K_1$, K2 . . . $K_{10}$ = Split (Dataset) // Divided Dataset into 10 parts
K_1[10] = {$K_1$, $K_2$ . . . $K_{10}$} // Storing all dividing indexes in K_1
i = 0 //Iterator
while (i < k)
    $Model_i$ = Decision tree (Hyperparameters)
    $Model_i$.train(Dataset -$K_i$)
    $Accuracy_i$ = $Model_i$.test($K_i$)
    i++
End
EnsembleModel = Maxvote($Model_1$, $Model_2$, . . . Model10)
Accuracies = {$Accuracy_1$, $Accuracy_2$ . . . $Accuracy_{10}$}
```

7 Results

The average accuracy obtained by the ensemble model based on the decision tree is 96.04% (see Fig. 2). The confusion matrix of the matrix explains on average how many samples were classified correctly for each class while testing (see Figs. 3, 4, 5, 6, 7, 8, 9, 10, 11 and 12).

Figure 13 plots graph between accuracy and *K* value, where *K* value determines the subset of data set as data set was divided into 10 equal parts. Highest accuracy 98.28%

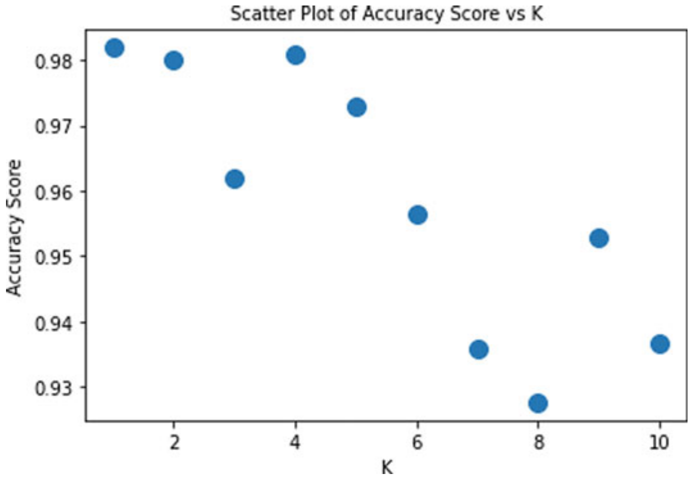
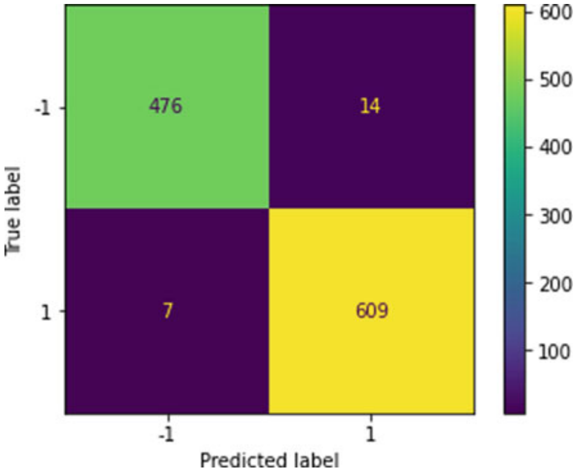


Fig. 2 Scatter plot of testing accuracy versus *K*

Fig. 3 Confusion matrix for 1st model



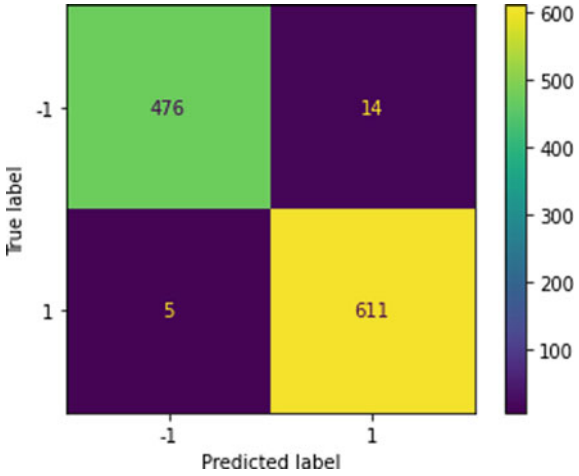


Fig. 4 Confusion matrix for 2nd model

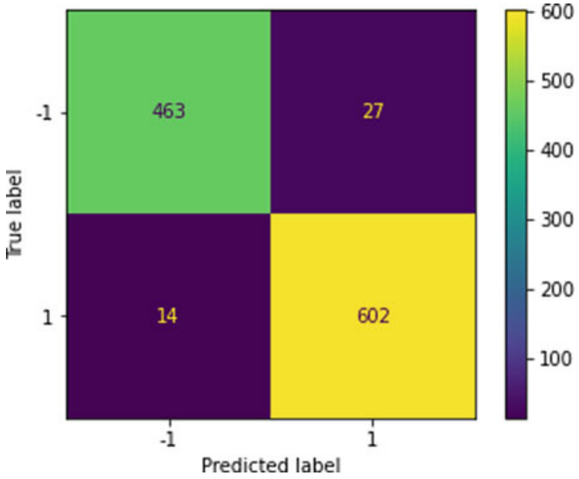


Fig. 5 Confusion matrix for 3rd model

was obtained on the fourth subset, while lowest accuracy 93.12% was obtained on eights subset. The average accuracy of the model is 96.04%.

Figure 14 plots graph between recall score and *K* value, where *K* value determines the subset of data set as data set was divided into 10 equal parts. Highest recall 0.9902 was obtained on the first subset, while lowest recall score 0.928 was obtained on seventh subset. The average recall score of the model is 0.967.

Figure 15 plots graph between precision score and *K* value, where *K* value determines the subset of data set as data set was divided into 10 equal parts. Highest precision 0.979 was obtained on the fourth subset, while lowest precision score 0.9206 was obtained on eight subset. The average precision score of the model is 0.961.

Fig. 6 Confusion matrix for 4th model

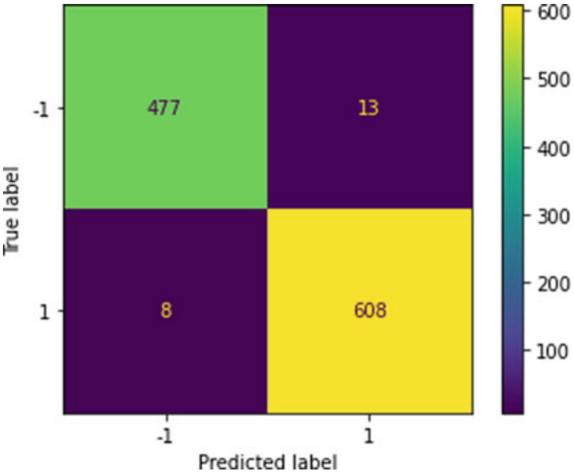


Fig. 7 Confusion matrix for 5th model

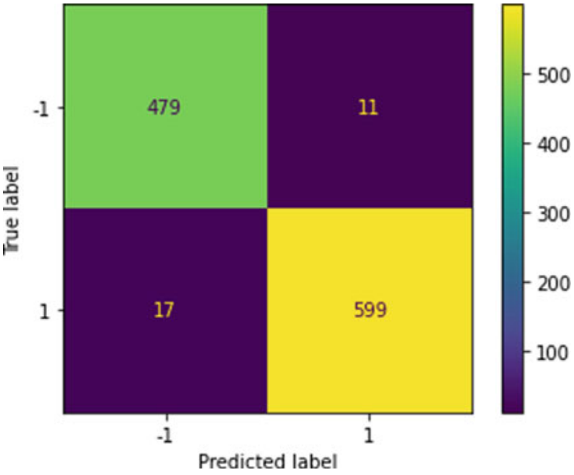


Figure 16 plots graph between $F1$ score and K value, where K value determines the subset of data set as data set was divided into 10 equal parts. Highest $F1$ score 0.984 was obtained on the fourth subset, while lowest $F1$ score 0.9387 was obtained on eighth subset. The average $F1$ score of the model is 0.964 All four precision, recall, $F1$ and accuracy scores have similar averages, and they all have identical graphs showing high capability of the ensemble model.

Fig. 8 Confusion matrix for 6th model

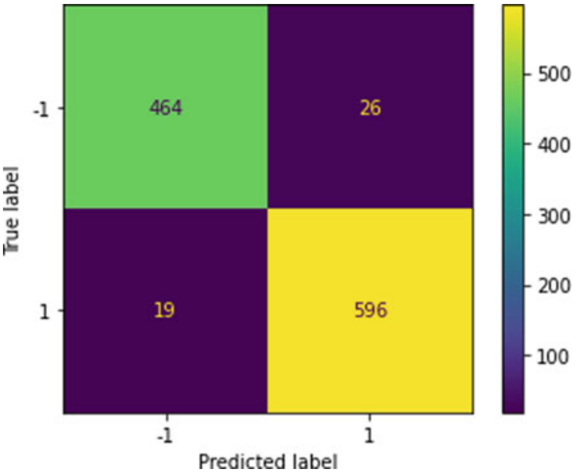
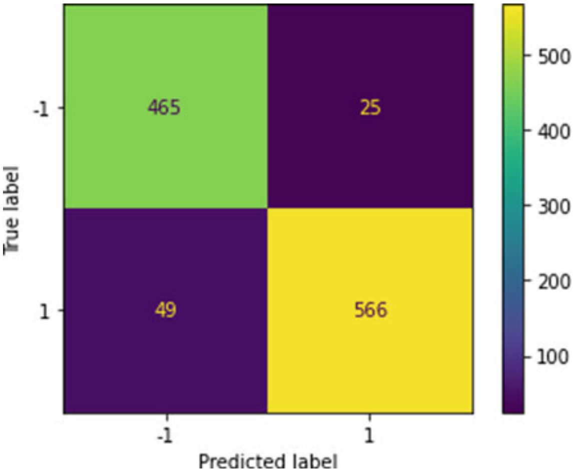


Fig. 9 Confusion matrix for 7th model



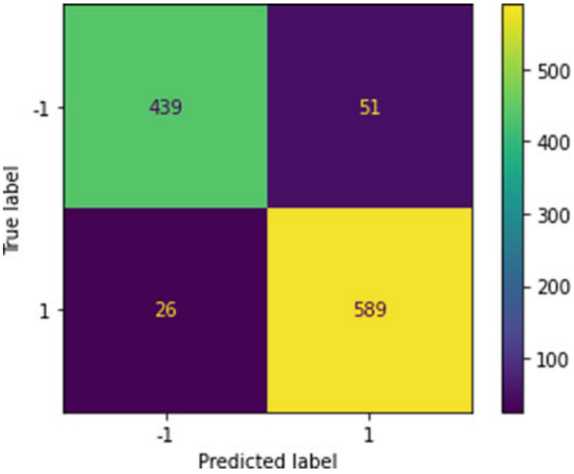


Fig. 10 Confusion matrix for 8th model

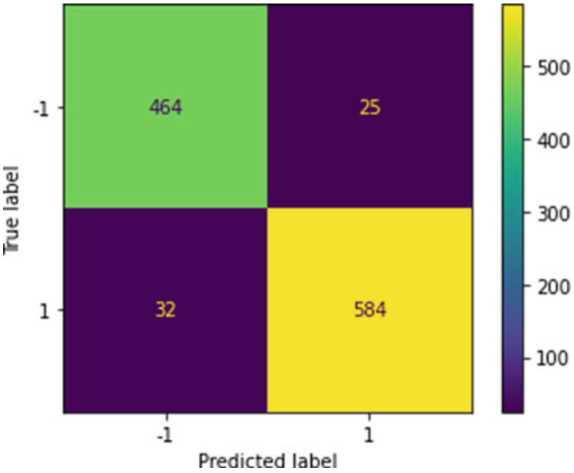


Fig. 11 Confusion matrix for 9th model

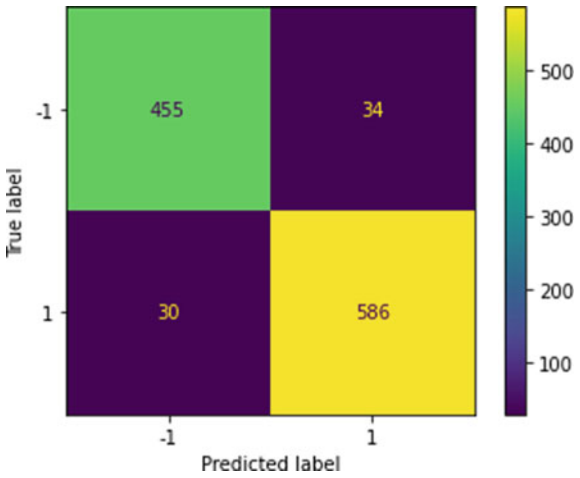


Fig. 12 Confusion matrix for 10th model

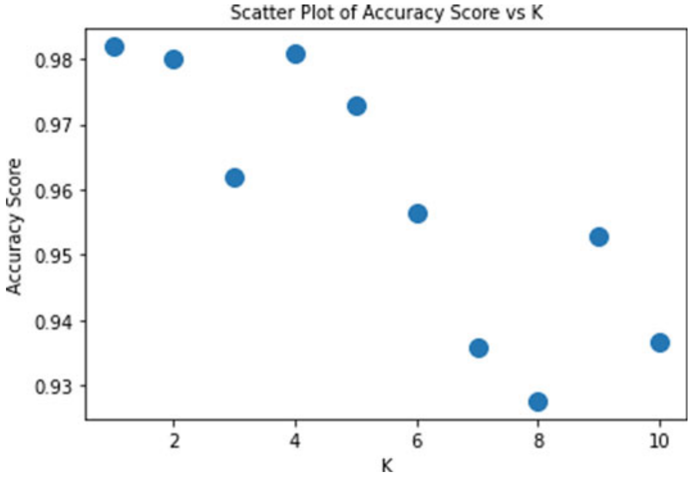


Fig. 13 Scatter plot of testing accuracy versus *K*

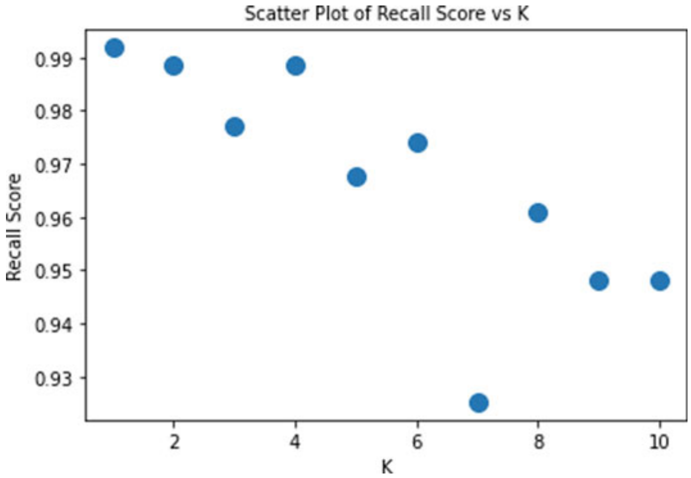


Fig. 14 Scatter plot of recall score versus *K*

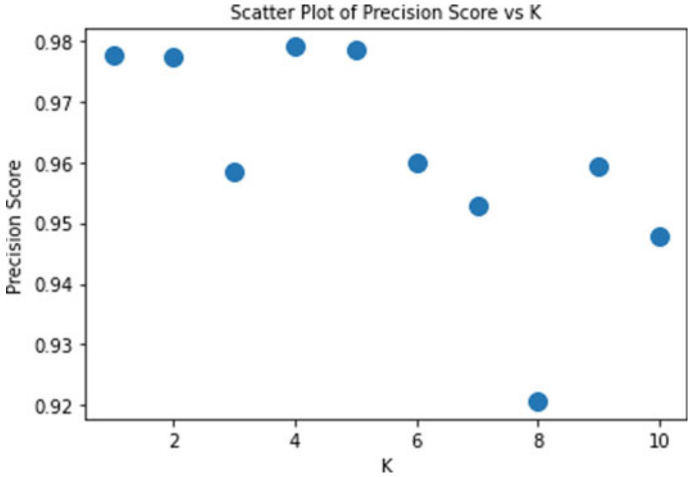


Fig. 15 Scatter plot of precision score versus *K*

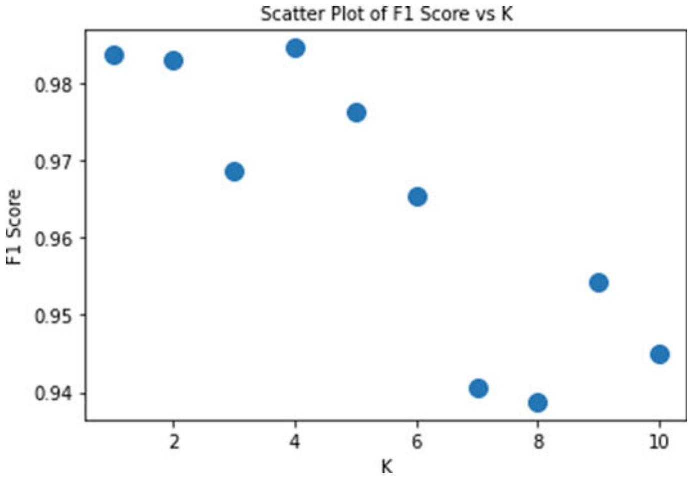


Fig. 16 Scatter plot of $F1$ score versus K

8 Conclusion

Phishing website scams have increased over the last one year due to the shift of mass population to digital life due to pandemic, and to tackle this problem, there is need of a phishing website detection software. Proposed ensemble model outshines all the previous existing research in comparison with their metrics.

The proposed model in this research was successful, and the objective of this research was achieved. This research can be used to implement a software, where user can input the URL of the website and on based of features detected from URL the software will classify whether website is phishing or not.

There is one shortcoming of this algorithm is that it requires high computational power in trade for the high accuracy as it takes 10 times more computational power in comparison to models created on single tree.

9 Future Scope

This research can be extended by providing a large data set of phishing websites, and in future, this research can be deployed in form of web application, so that it can be used by any user to consult this application about website before entering their sensitive information on that website.

References

1. Mahajan R, Siddavatam I (2018) Phishing website detection using machine learning algorithms. Int J Comp Appl 181(23):45–47. https://doi.org/10.5120/ijca2018918026
2. Mehanović D, Kevrić J (2020) Phishing website detection using machine learning classifiers optimized by feature selection. Traitement Du Sign 37(4):563–569. https://doi.org/10.18280/ts.370403
3. Geyik B, Erensoy K, Kocyigit E (2021) Detection of phishing websites from URLs by using classification techniques on WEKA. In: 2021 6th International conference on inventive computation technologies (ICICT). Published. https://doi.org/10.1109/icict50816.2021.9358642
4. Manoj P, Bhuvan Kumar Y, Rakshitha D, Megha G (2021) Detection and classification of phishing websites. Trends Comp Sci Inform Technol 053–059. https://doi.org/10.17352/tcsit.000040
5. Mourtaji Y, Bouhorma M, Alghazzawi D, Aldabbagh G, Alghamdi A (2021) Hybrid rule-based solution for phishing URL detection using convolutional neural network. Wirel Commun Mob Comput 2021:1–24. https://doi.org/10.1155/2021/8241104
6. Pati Burman P (1989) A comparative study of ordinary cross-validation, v-fold cross-validation and the repeated learning-testing methods. Biometrika 76(3):503. https://doi.org/10.2307/2336116
7. Natadimadja MR, Abdurohman M, Nuha HH (2020) A survey on phishing website detection using Hadoop. Jurnal Informatika Universitas Pamulang 5(3):237. https://doi.org/10.32493/informatika.v5i3.6672

Prediction of Trait Anxiety in Humans

Tiya Kahai, Paarth Modgil, Ms Kavita, and Rahul Saxena

Abstract Concerns about mental health have been increasingly common in recent decades. Mental health problems have existed from the beginning but have always been stigmatized by people. According to statistics, one out of every four persons in the world today has a mental health problem and despite this, only a few people receive mental health assessment or treatment. Anxiety is one of the most common psychological emotions felt and is often stated to be healthy but becomes a medical disorder when a person regularly feels disproportionate levels of it. The aim of this study is to predict the trait anxiety in humans at an early stage so that it can be acted upon accordingly by professionals and lead to a better diagnosis and treatment of the subject. We have implemented machine learning algorithms to analyze data and produce a highly accurate model. We have used the TMAS or Taylor Manifest Anxiety Scale Response as dataset, which is a response questionnaire that consists of 50 True and False questiossns and predicts anxiety in humans based on the test score obtained by the subject. The test is scientifically backed and used extensively in the field of psychiatry bearing positive outcomes. The machine learning algorithms used were support vector machine (SVM), random forest (RFA), and K-nearest neighbor (KNN). Highest accuracy was found by using the support vector machine (SVM) algorithm.

Keywords Anxiety · Mental health · Machine learning · Support vector machine · Random forest · K-nearest neighbor

1 Introduction

Anxiety is the most common psychological illness today. The World Health Organization (WHO) projects that about 7.5% of Indians suffer from a mental health illness, and that by the end of 2020, roughly 20% [9] of Indians will be suffering. Statistics show that 56 million Indians suffer from depression and 38 million suffer from

T. Kahai · P. Modgil · M. Kavita (✉) · R. Saxena
Department of Information Technology, Manipal University Jaipur, Jaipur, Rajasthan, India
e-mail: kavita.chaudhary@outlook.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_49

anxiety disorders. Sigmund Freud was the first to scientifically define the human anxious psychophysiological response as a sense of imminent and pressing danger that could be based on objective or moral risk [1, 2]. Anxiety is defined as an acute feeling of tension, stress, or dread about the future. However, if anxiety becomes severe and uncontrolled, it can become problematic, opening the door for anxiety disorders to emerge. Psychiatry has long sought diagnostics that can scientifically diagnose patients, assess therapy response, or define illness risk. Reliable psychological biomarkers, on the other hand, have yet to be discovered. Machine learning methods have recently been used to develop biomarkers, with promising preliminary results [3].

In this study, we propose to predict anxiety in humans by using machine learning algorithms. We will be using publicly available data on Kaggle, Taylor Manifest Anxiety Scale Response Data. This analysis is bound to have multiple applications in various fields ranging from that of medical diagnostics to increasing awareness. Hence, this study could resolve multiple existent problems that humankind is facing and find multiple new applications as and when the need for the same comes up [4, 5].

2 Literature Review

Many studies have presented the relevant work predicting anxiety using machine learning algorithms, such as random forest, support vector machine, and K-nearest neighbor. Sau et al. [6] gathered primary data from Medical College and Hospital of Kolkata, India of aged people, some of them were in special care. After implementing several classification methods, Bayesian network, logistic, multiple layer perceptron, Naïve Bayes, random forest, random tree, J48, sequential random optimization, random sub-space, and K star. The results of the study show that random forest performed better than other methods with 91% and 89% accuracy rate for two datasets, respectively. Priya et al. [5] predicted anxiety, depression, and stress using decision tree, random forest, support vector machine, Naïve Bayes, and K-nearest neighbor. The data was collected independently using depression, anxiety, and stress scale questionnaire (DASS 21). The study revealed that random forest was the best performing method among all the methods with 71% accuracy rat.

In study, multiple machine learning algorithms were applied, such as logistic regression, CatBoost, Naïve Bayes, RFT, and SVM for classification. The researchers [7] interviewed many seafarers to collect the data via 16 characteristics. The study found that CatBoost gave highest level of precision and accuracy of 82.6% and 84.1%, respectively, among all classifiers. The authors [8] presented neural network model for the prediction of evolving psychological conditions for instance anxiety, behavioral disorders, depression, and post trauma stress. The dataset contained 89,840 patients' information for the analysis. The results of the study shown that the achieved accuracy may range from 73 to 95% for all the clinical situations, along with the accuracy of 82.35%.

3 Methodology

Dataset

For our study, we used the publicly available 'Taylor Manifest Anxiety Scale Response Data' on the Kaggle. Responses were collected from 5410 subjects. To avoid oversampling, we underfitted our data and trained and tested our model on 2088 subjects. The data was first made available on OpenPsychometrics.org (This is a non-profit organization dedicated to public education and data collection for psychological study and research) [9].

The Taylor Manifest Anxiety Scale (TMAS) is a questionnaire-based assessment that assesses anxiety as a personality trait. It was established by Janet Taylor in 1953 with the purpose of selecting volunteers who would be useful in anxiety disorder research. The TMAS consists of 50 True or False questions that a person answered by introspection to determine their anxiety level. During her career as a psychologist, Janet Taylor focused on anxiety and gender development. Her scale has been widely used to distinguish between normal people and test subjects with pathological anxiety levels. The Taylor Manifest Anxiety Scale has been shown to have test–retest reliability and has been used in the field of psychiatry bearing positive outcomes.

The answers of the questionnaire was rated as 1 = True, 2 = False and 0 = not answered. If a subject scores 25 or more points in the above question test, they are supposed to be suffering from any level of anxiety and need immediate medical assistance and mental support.

Classification

Three machine learning techniques were used to predict different levels of accuracy. The algorithms applied here are random forest algorithm, KNN algorithm, and support vector machine algorithm. After supervised learning on the subject's answers to a TMAS questionnaire, the model with the best accuracy was chosen for future real-time application. Figure 1 shows the flow of the study.

A) Random Forest Algorithm (RFA)

   For supervised learning, the random forest algorithm is utilized (supervised learning is a process where the user provides the machine learning model with input as well as the correct output data, and then model is trained using correctly labeled training data). Using the "bagging" method, it creates a "forest" out of a collection of decision trees that are generally trained. Mixing several learning models improves the overall analytic result. The more the trees in the forest, the more accurate the outcome, and the problem of data overfitting is likewise resolved.

B) K-Nearest Neighbor Algorithm (KNN)

   Based on this assumption, the K-nearest neighbor technique allocates the current instance to the category that is closest in similarity with previous examples.

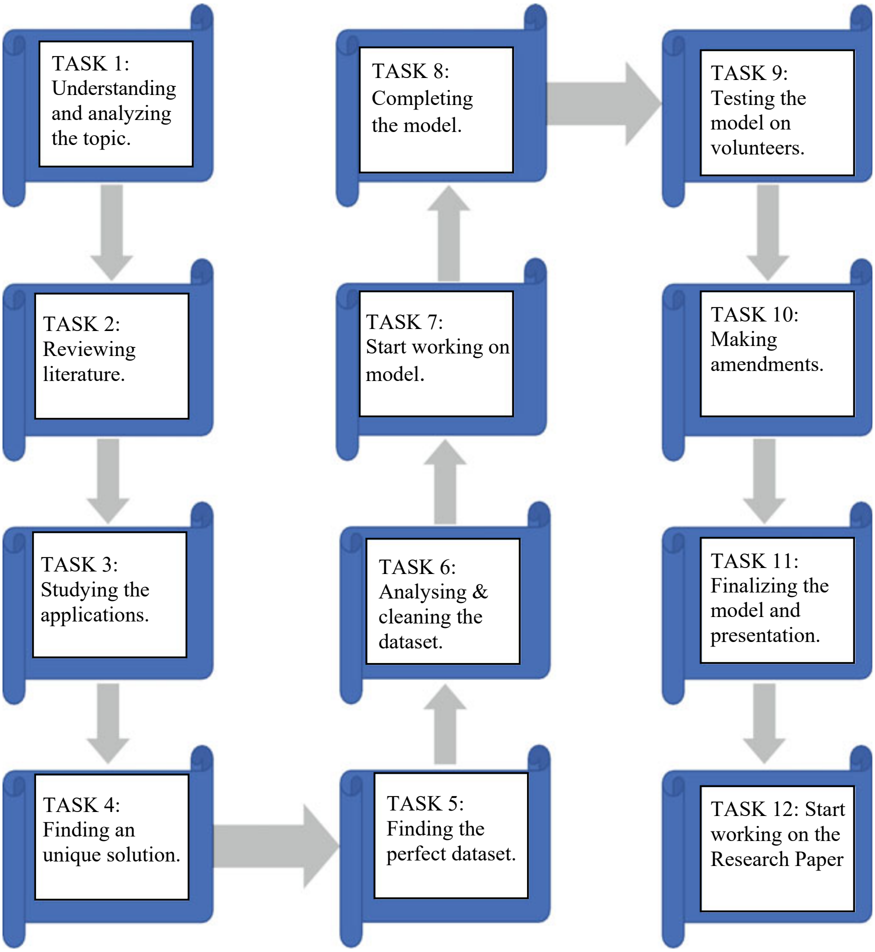

TASK 1: Understanding and analyzing the topic.
TASK 2: Reviewing literature.
TASK 3: Studying the applications.
TASK 4: Finding an unique solution.
TASK 5: Finding the perfect dataset.
TASK 6: Analysing & cleaning the dataset.
TASK 7: Start working on model.
TASK 8: Completing the model.
TASK 9: Testing the model on volunteers.
TASK 10: Making amendments.
TASK 11: Finalizing the model and presentation.
TASK 12: Start working on the Research Paper

Fig. 1 Flowchart of the study

According on how similar fresh data points are to existing data, the KNN algorithm saves all data. This indicates that the nearest neighbor algorithm can swiftly classify new data into a well-defined category.

C) Support vector machine (SVM)

The support vector machine's purpose is to identify the ideal classification line or decision border to classify n-dimensional space, making it easier to identify and classify new data points. The best judgement boundary is also known as a hyperplane. SVM itself chooses the hyperplane's extreme vectors. In the SVM algorithm, a decision boundary or hyperplane is used to classify two different groups.

There can be multiple lines/choice boundaries to split the groups in n-dimensional space, but we need to determine the optimum decision boundary to help characterize the data points. The hyperplane of SVM refers to the best boundary. The data points or vectors that are closest to the hyperplane and impact the hyperplane's location are known as support vectors. These vectors are called support vectors since they support the hyperplane. SVM is further divided into:

1. Linear SVM
2. Nonlinear SVM

4 Result and Discussion

We first started by doing the exploratory analysis of the data and visualized it to get a better understanding of how to use our data. We intensively cleaned our data for the most accurate results. Our data consisted of 53 columns: total score, gender, age, and the score of each of the 50 questions.

Our data was oversampled, so we underfitted our data, as depicted in Figs. 2 and 3. The data has been visualized for the better analysis through correlation matrix, distribution graphs, and confusion matrices (Figs. 4, 5, 6, 7 and 8).

Then we applied the machine learning algorithms on our data with different splits, with the following results.

Terminology

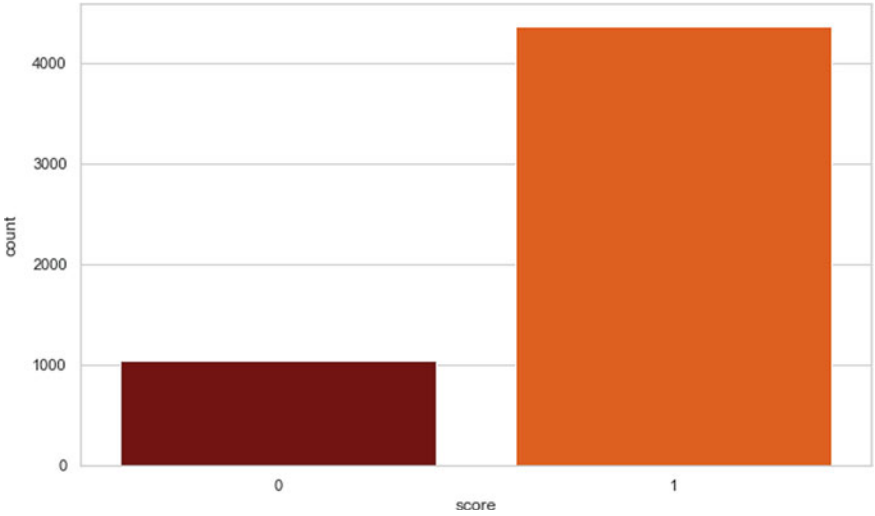


Fig. 2 Oversampled data

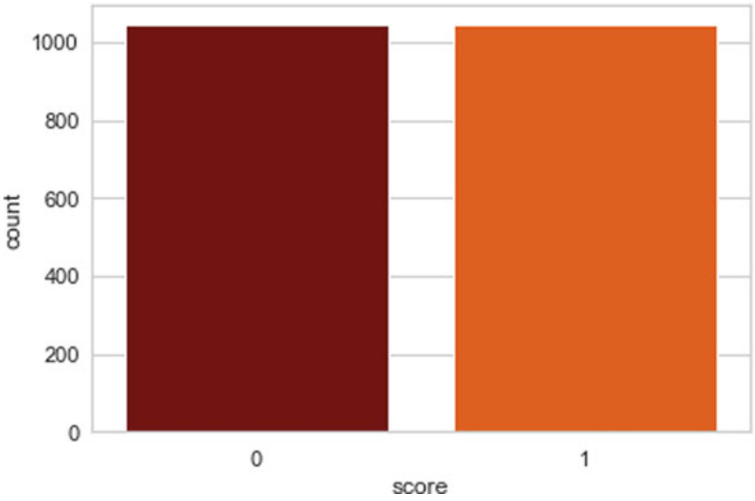


Fig. 3 Underfitted data

	Predicterd class		
Actual class		Class = yes	Class = no
	Class = yes	**True positive**	*False negative*
	Class = no	*False positive*	**True negative**

Accuracy

$$\text{Accuracy} = \text{TP} + \text{TN}/\text{TP} + \text{FP} + \text{FN} + \text{TN}$$

Precision

$$\text{Precision} = \text{TP}/\text{TP} + \text{FP}$$

Recall

$$\text{TP}/\text{TP} + \text{FN} = \text{Recall}$$

$$F1\ \text{Score} = 2 * (\text{Recall} * \text{Precision})/(\text{Recall} + \text{Precision})$$

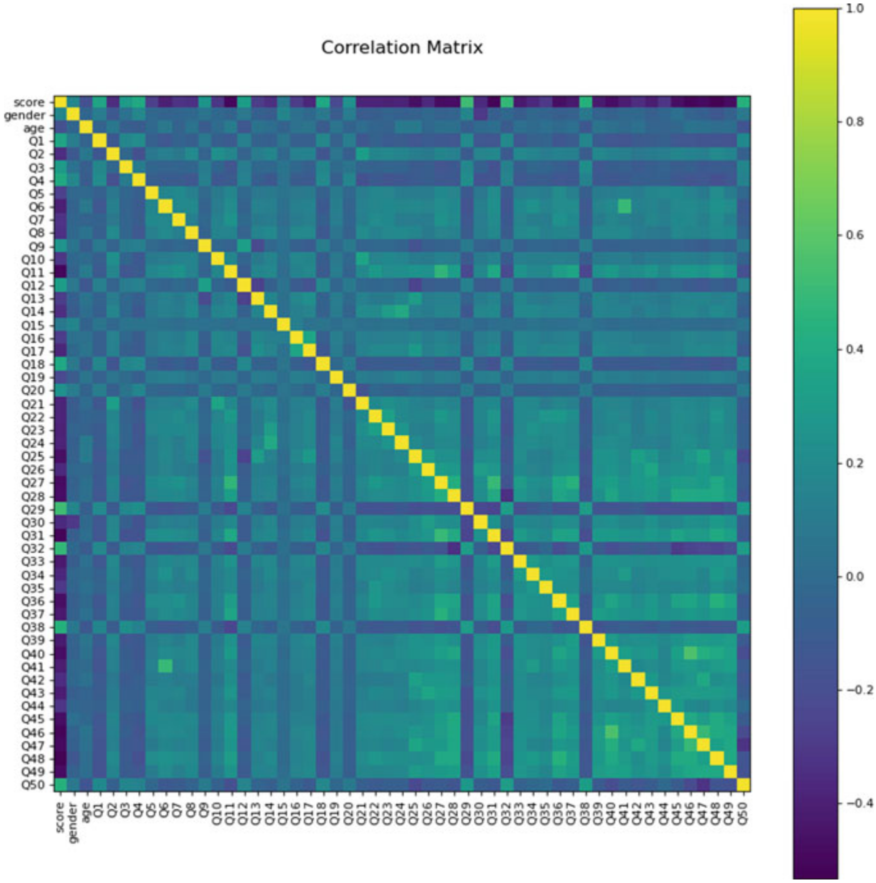


Fig. 4 Correlation matrix

Final comparison of the accuracies is as shown in Fig. 9.

Hence, using the most accurate algorithm, i.e., SVM algorithm, we have created our model to predict anxiety at early stages, so that subjects get proper medical attention at an early stage.

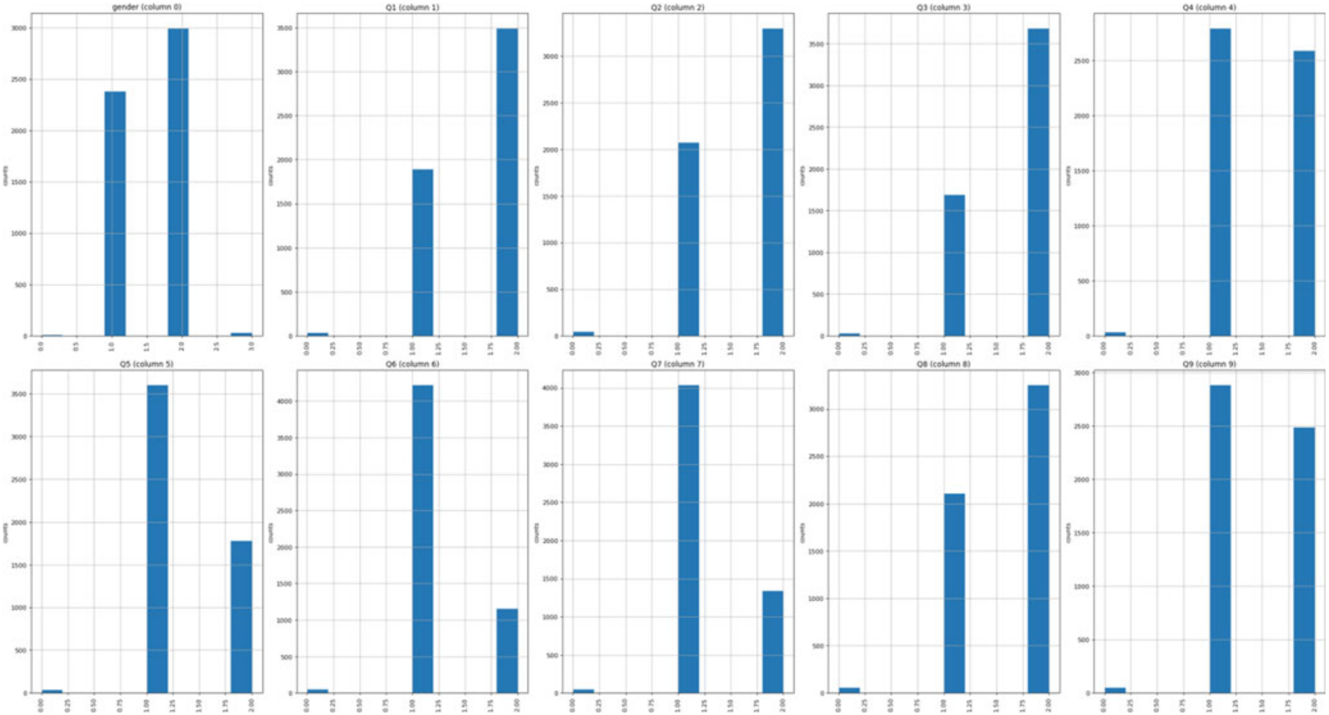
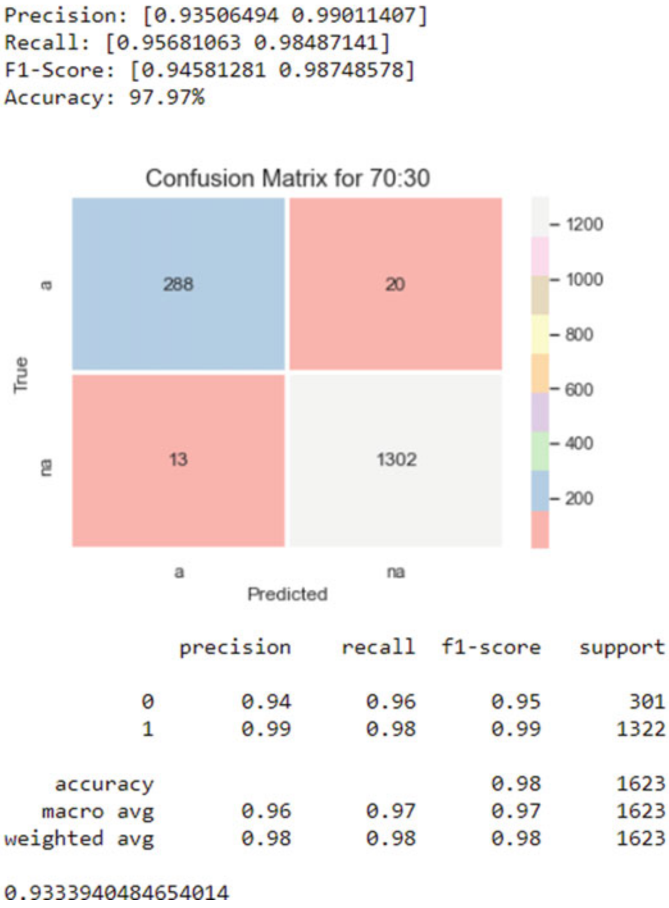


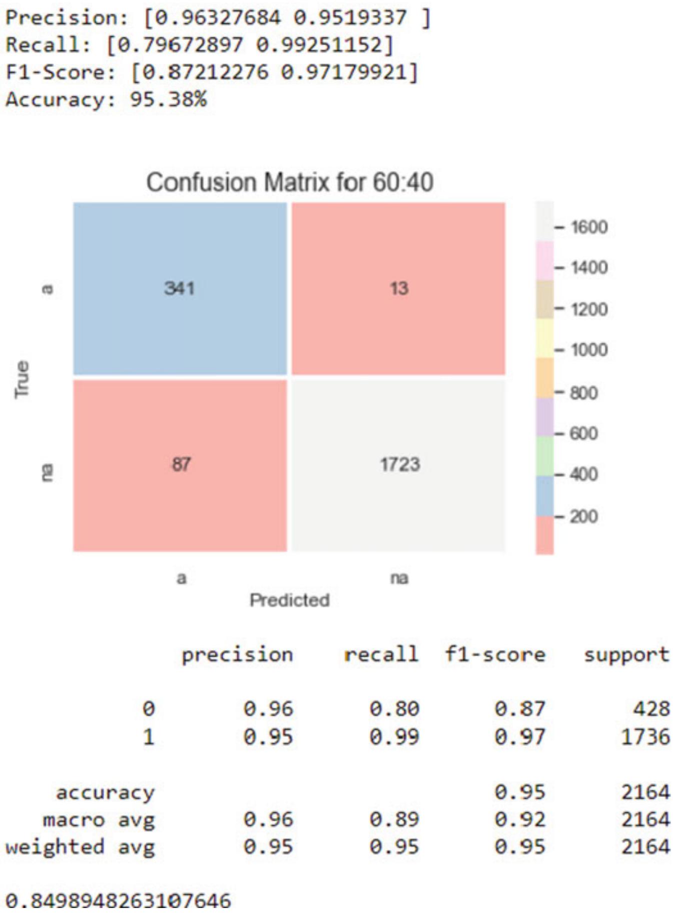
Fig. 5 Distribution graphs



```
Precision: [0.93506494 0.99011407]
Recall: [0.95681063 0.98487141]
F1-Score: [0.94581281 0.98748578]
Accuracy: 97.97%
```

```
              precision    recall  f1-score   support

           0       0.94      0.96      0.95       301
           1       0.99      0.98      0.99      1322

    accuracy                           0.98      1623
   macro avg       0.96      0.97      0.97      1623
weighted avg       0.98      0.98      0.98      1623

0.9333940484654014
```

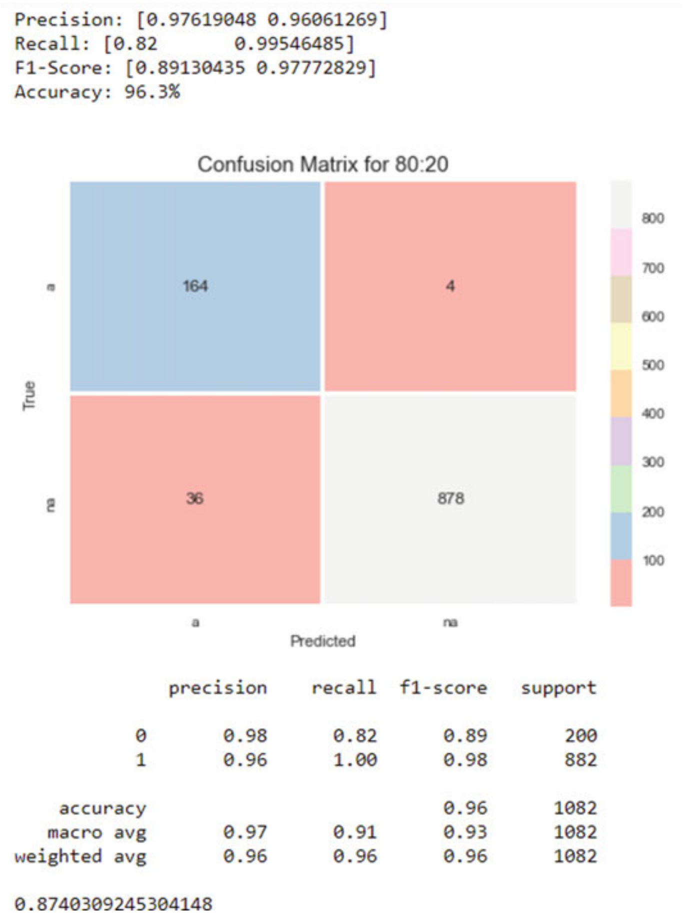
Fig. 6 SVM



```
Precision: [0.96327684 0.9519337 ]
Recall: [0.79672897 0.99251152]
F1-Score: [0.87212276 0.97179921]
Accuracy: 95.38%
```

```
              precision    recall  f1-score   support

           0       0.96      0.80      0.87       428
           1       0.95      0.99      0.97      1736

    accuracy                           0.95      2164
   macro avg       0.96      0.89      0.92      2164
weighted avg       0.95      0.95      0.95      2164

0.8498948263107646
```

Fig. 7 KNN



```
Precision: [0.97619048 0.96061269]
Recall: [0.82       0.99546485]
F1-Score: [0.89130435 0.97772829]
Accuracy: 96.3%
```

```
              precision    recall  f1-score   support

           0       0.98      0.82      0.89       200
           1       0.96      1.00      0.98       882

    accuracy                           0.96      1082
   macro avg       0.97      0.91      0.93      1082
weighted avg       0.96      0.96      0.96      1082

0.8740309245304148
```

Fig. 8 Random forest

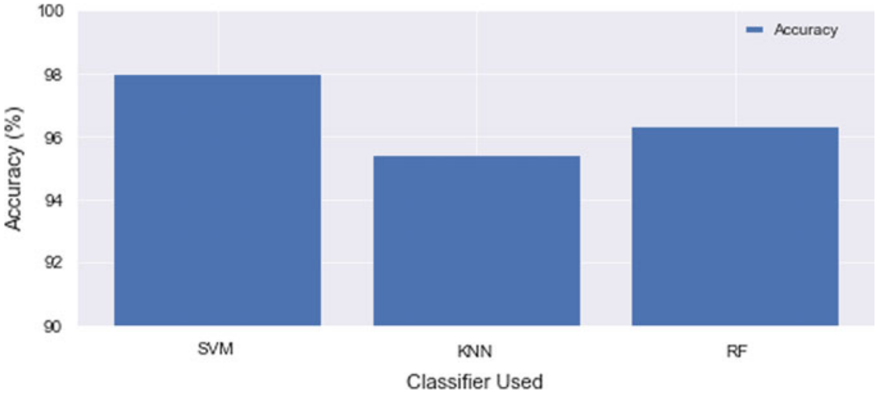


Fig. 9 Comparison of the accuracies

5 Conclusion and Future Scope

The goal was to detect the development of trait anxiety in people at an initial stage through the investigation of several underlying factors. This study used three different machine learning models to predict the severity of anxiety. These methods are classified into: SVM, KNN, and random forest. After implementing all the techniques, the results revealed that the SVM algorithm outperformed all the others. With people silently suffering with anxiety and not even addressing the underlying issue because of unreal societal norms is what motivated us to work on this paper. We want people around us to break out of the societal pressure and stigma. In conclusion, our paper has several applications and if implemented correctly, we can detect multiple mental health issues at an early stage. If we all use technology the right way, we can use it to our huge advantage and help people in need. We further plan to explore different mental health conditions and study if it is possible to detect such diseases using machine learning models.

References

1. Richter T, Fishbain B, Markus A, Richter-Levin G, Okon-Singer H (2020) Using machine learning- based analysis for behavioral differentiation between anxiety and depression. Sci Rep 10(1):16381. https://doi.org/10.1038/s41598-020-72289-9
2. Bokma WA, et al (2020) Predicting the naturalistic course in anxiety disorders using clinical and biological markers: a machine learning approach. Psychol Med 1–11. https://doi.org/10.1017/S0033291720001658
3. Boeke EA, Holmes AJ, Phelps EA (2020) Toward robust anxiety biomarkers: a machine learning approach in a large-scale sample. Biol Psychiatry Cogn Neurosci Neuroimaging 5(8):799–807. https://doi.org/10.1016/j.bpsc.2019.05.018

4. Kumar P, Garg S, Garg A (2020) Assessment of anxiety, depression and stress using machine learning models. Procedia Comput Sci 171:1989–1998. https://doi.org/10.1016/j.procs.2020.04.213
5. Priya A, Garg S, Tigga NP (2020) Predicting anxiety, depression and stress in modern life using machine learning algorithms. Procedia Comput Sci 167:1258–1267. https://doi.org/10.1016/j.procs.2020.03.442
6. Sau A, Bhakta I (2017) Predicting anxiety and depression in elderly patients using machine learning technology. Healthc Technol Lett 4(6):238–243. https://doi.org/10.1049/htl.2016.0096
7. Sau A, Bhakta I (2019) Screening of anxiety and depression among the seafarers using machine learning technology. Inform Med Unlocked 16:100149. https://doi.org/10.1016/j.imu.2018.12.004
8. Dabek F, Caban JJ (2015) A neural network based model for predicting psychological conditions. In: Brain informatics and health, Cham, pp 252–261. https://doi.org/10.1007/978-3-319-23344-4_25
9. “Manifest Anxiety Scale Responses.” https://kaggle.com/lucasgreenwell/manifest-anxiety-scale-responses. Accessed 03 Aug 2021

Deep Learning-Based Object Detection: An Investigation

Kanojia Sindhuben Babulal and Amit Kumar Das

Abstract Computer vision has one most important and challenging problem of object detection because of its wide application in field of deep learning such as medical image analysis and security monitoring autonomous driving. Object detection tasks have been greatly improved as object detection has compact association with video evaluation and image processing, and it has enticed the notice of researchers in adjunct years and describe the reference datasets at the beginning. This paper provides a complete review of a range of object detection technique, in a structured way detailing about the two-stage and one-stage detector, including the algorithms used both in detectors and in R-CNN, fast R- CNN and faster R-CNN. R-CNN, YOLO, SSD mask, etc. Also, we list the traditional (only detect object and its type) and new app (object detection with analysis and learning). Few indicative divisions of object detection are also discussed, and eventually, the performance of all models used in a one-stage and two-stage detector is discussed. We too in short examine various distinct jobs, together with projecting, face detection, object detection and pedestrian detection. Finally, various budding orientation and trends are furnished that assist as challenges or recommendation for upcoming prospective job.

Keywords Object detection · Deep learning · Computer vision · R-CNN · Fast R-CNN · Faster R-CNN · Mask R-CNN · YOLO · SSD

1 Introduction

Today's techniques for detection of objects have attracted more and more recognition in current years because of its broad variety of applications and current hi-tech advancements. This assignment is under expansive examination, inspection or real-world application for instance security reconnoiter, independent driving, transport surveillance, drone scene investigation and robotic vision. At the moment, the deep

K. S. Babulal (✉) · A. K. Das
Department of Computer Science and Technology, Central University of Jharkhand, Ranchi, Jharkhand, India
e-mail: sindhukanojia@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_50

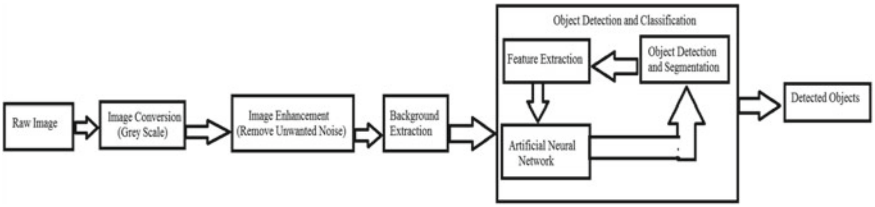

Raw Image
Image Conversion (Grey Scale)
Image Enhancement (Remove Unwanted Noise)
Background Extraction
Object Detection and Classification
Feature Extraction
Object Detection and Segmentation
Artificial Neural Network
Detected Objects

Fig. 1 Object detection workflow

learning model is broadly adopted throughout the area of computer vision, together with popular object detection as well as domain-specific object detection. Latest generation object detectors use detection and in background network utilize deep learning to extract characteristics out of provided images or videos, classification along with location separately. Object detection is an advance computer mechanism associated to image processing and computer vision in such a way that explores along with identifying occurrence of meaningful objects of a hundred of classes like as humans, cars, animal, buildings, etc. Well-investigated domains of object detection in digital images and videos comprise multi-category detection, edge detection as well as protruding object, pose, scene text, face detection, etc. [1]. Object detection is commonly applied in variety of domains of present life for instance security area, military area, transportation area, medical area and area of the life as a principal part of scene understanding. Furthermore, many milestones have played a vital part in the object detection field until now. Figure 1 illustrates the glimpse of work flow of object detection. This survey paper describes the use of different models and algorithm used for object detection and their performance including their working. And how object detection will help to build such application which can facilitate service to the human being.

2 Kind of Object Detector

Models in object detection approach is widely classified in two broad types (a) single-stage detector like YOLO and SSD and (b) two-stage detectors, for example, faster R-CNN, etc. Two-stage detector attains highest object detection accuracy but is slow while single stage detector highlights on inference speed and are very fast. Faster R-CNN (two-stage detector) executes in two steps, where first step builds RPN (Region Proposal Network) which produce region proposal that is provided to object detection model. In the second phase, the characteristics are obtained using the RoIPool (RoI grouping) functioning from every candidate box for the upcoming classification tasks as well as bounding box regression. Fundamental framework of two-stage detectors is exhibited in Fig. 2. Additionally, box prediction from the input images with the help of single-stage detector can be done without proposal of step region making SSD extra time efficacious and further may be implemented in real-time appliances such as live monitoring via Raspberry Pi and vision by computer. Figure 3 represents the architecture of one-stage detector.

The primary benefaction of this work is recapitulated as follows: This research paper focuses on narrating and examining object detection technique based on deep learning in which object detection is major concern. This survey highlights on limited areas of object detection and do not include intricate method that might provide extra ordinary answers to progress of the computer vision analysis.

Here, the survey presents thorough study and discussion in several aspects, to the best of our understanding are the newly introduced in this area. Furthermore, non-identical from preceding surveys on object detection, this research article present methodical as well as diverse report on deep learning-based object detection technique and very crucial the modern detection results and group of notable research

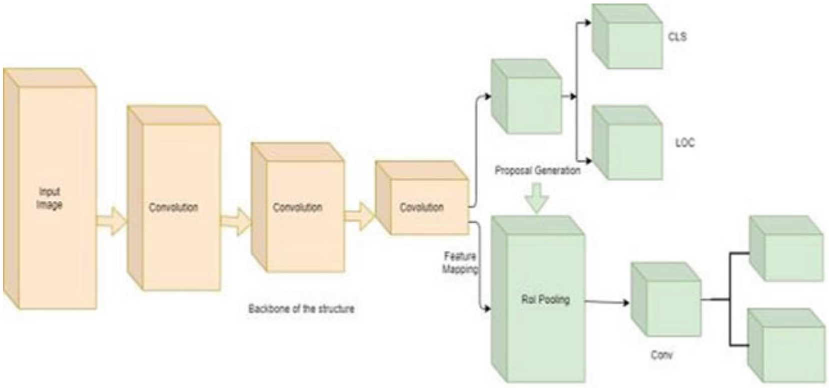


Fig. 2 Display the fundamental framework of two-stage detector that comprise region proposal network to supply region proposal within classifier plus regressor

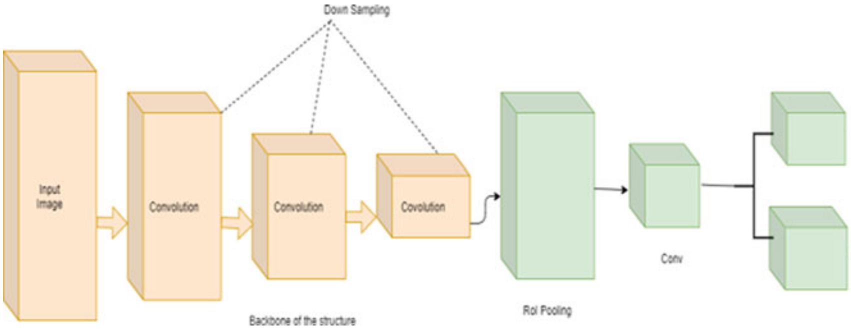


Fig. 3 Represent the fundamental framework of one-stage detector, predicting bounding boxes from provided input images instantly

inclination too. So, in this paper, our objective is to furnish an outline of how divergent deep learning techniques are utilized instead of a whole outline of all associated research articles. For better understanding of this topic, we endorse readers refer to [1, 2, 3].

The remaining of the paper is organized as depicted: Sect. 3 explains the backbone network as we know strong backbone network is important to extricate abundant attributes. Section 4 deals with the dataset; Sect. 5 discusses in detail about the applications and branches; Sect. 6 talks about the results, and Sect. 7 ends with conclusions and trends.

3 Backbone Network

Normally, domain-specific imaging object detectors are divided into two groups: one-stage detector, for example, YOLO, SSD, and other is two-stage detector, for example, faster R-CNN. High precision of location and object identification, while the one-stage detector attains elevated speed of inference. And two-stage detectors have two stages which can be divided by the grouping layer of region of interest (ROI), for example, first stage within faster R-CNN is termed RPN, which suggests frames candidate object delimiters. In the next stage, from ever single candidate box features are extracted in upcoming classification along with bounding box regression jobs using the ROIPOOL operation (ROI pooling). Figure 2 exhibits the primary framework of two-stage detectors. In addition, one-stage detectors advocated anticipated boxes out of input images straight way lacking a region suggestion phase. Therefore, these approaches are time saving as well as implemented with real-time devices, e.g., for live monitoring by Raspberry Pi and computer vision. Figure 3 exhibits primary framework of single-stage detectors. This detector was developed as backbone to accommodate detection layout but these are poor at identifying uneven shaped objects or a batch of small objects. The newly developed highly efficient classification networks can enhance the application as well as diminish the complication of the object detection mission. This is an effectual procedure to further improve network performance due to the backbone network acting as a feature extractor.

3.1 Typical Baseline

With expeditious growth of deep learning and the side-by-side upgradation of computing power, an enormous advancement has been done through area of normal object detection. Accompanied with the progress of earliest CNN-based object detector, R-CNN was introduced, and a sequence about notable advancement is carried that encourages expansion of common object detection with a massive limit. For beginners, some representative object detection framework has been initiated to obtain commencement within said field.

3.2 Two-Stage Detector

R-CNN: Region-based detector (R-CNN) has four modules. Category-independent region is produced by first module region, while the second module separates each region proposal from fixed-length feature vector. The third section acts as a group of class-specific linear SVMs for categorizing the objects present inside single image, and fourth and last section showcases bounding-box regressor to accurately predict bounding box. This uses prior-training on huge dataset accompanied by adjustment on particular dataset is a commendable technique to reach rapid convergence.

Fast R-CNN: The drawback of R-CNN taking long processing time on SVM's classification leads Ross Girshick [4] to propose a speedy variant of R-CNN, termed as fast R-CNN after one year of R-CNN. Fast R-CNN separates features out of a complete photograph as input as well as subsequently proceeds the region of interest (ROI) pooling layer to acquire the stilled size characteristics serving as input toward classification along with bounding box regression are completely connected layers. The feature is separated by the complete image one time and as well as is forwarded to CNN toward classification and localization. While in R-CNN complete inputs image every region is forwarded to CNN that consumes excess time which is stored in fast R-CNN because CNN's processing time and huge disk depot to accumulate a prominent trade in features is saved.

Faster R-CNN: Faster R-CNN [5] proposed that after 3 months of fast R-CNN additionally enhances the region-based CNN baseline. Selective search is utilized by fast R-CNN to recommend region of interest that is indeed stolid as well as requires the identical execution time as the detection network. RPN is substituted in faster R-CNN that is completely convolutional network to accurately anticipate region proposal along with a vast scope of scales as well as aspect ratios. RPN improves speed of region proposals generation because it dispenses full-image convolutional characteristics along with a general group of convolutional layers including the detection network. Assessment demonstrates precision along with detection efficiency exceeds in faster R- CNN. Fast R-CNN outperforms faster R-CNN 10 times in total running time with identical backbone.

Mask R-CNN: This is expansion of faster R-CNN basically designed for instance segmentation task. Mask R-CNN is more precise object detector. He et al. utilize faster R-CNN accompanied by ResNet [6]-FPN [7] (feature pyramid network, a backbone separates ROI characteristic by various level about the feature pyramid relative for its scale).

3.3 One-Stage Detector

YOLO: You Only Look Once (YOLO) was introduced by Redmon et al. [8], which predicts both together confidence for numerous groups as well as bounding boxes. Working of You Only Look Once is displayed in Fig. 4. YOLO splits the given image within S × S grid; moreover every grid cell remains accountable toward estimating the object focused within particular grid cell. Every grid cell estimates B bounding boxes along with their equivalent confidence scores. This confidence scores actually interpreted as Pr(Object) * IOU pred$^{\text{truth}}$ that demonstrate how probably there occur objects (Pr(Object) > = 0) as well as present confidence of its prediction (IOU$^{\text{truth}}$pred).

There are 24 convolutional layers (CONV) including two fully connected layers (FC) in YOLO. Few CONV layers devise groups of initiation section with 1*1 reduction layers accompanied through 3*3 CONV layers. Speed in which network could operate on images in real time at 45 frame per second (FPS) as well as fast YOLO extends by 155 FPS with preferable outcomes as compared to remaining contemporaneous detectors. Moreover, YOLO construct scares false positives on backdrop that helps in coordination through fast R-CNN. Later an advanced edition, YOLOv2 was suggested in [9], which supports various magnificent techniques for instance (batch normalization) BN, anchor boxes, and dimension cluster including multi-scale training.

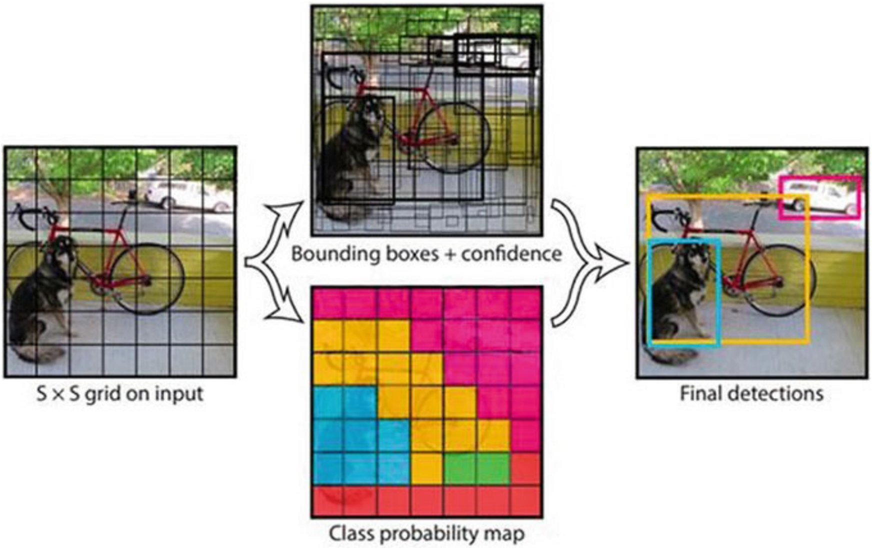


Fig. 4 Working of YOLO [8]

YOLOv2: [10] is advanced edition of YOLO [11] that makes arrogate sequence of blueprint conclusion by previous tasks with novel idea to enhance YOLO's precision and speed.

YOLOv3: It is an enhanced edition of YOLOv2. Initially, YOLOv3 utilize multi-label classification to accommodate additional difficult, dataset comprising overlapping labels. YOLOv3 utilizes three distinct scale feature maps to forecast the bounding box. 3-D tensor encoding class prediction, objectiveness and bounding boxes are forecasted by the extreme convolutional layer. YOLOv3 recommends an in depth as well as vigorous feature extractor, named darknet-53, influenced through ResNet.

SSD: YOLO is strenuous in tackling small-sized objects in set because of strong spatial constraints forced on bounding box predictions [8]. At the same time, YOLO scuffles to approximate to objects in current/uncommon aspect ratios/layout and assemble, respectively, substantial features because of numerous down sampling functioning.

Liu at el. discuss a single-shot multi-box detectors (SSD) [12], motivated by the anchors embraced in multi-Box [13], RPN [4] including multi-scale portrait [14]. Provided particular feature map, SSD captures the benefit concerning group of conventional anchor boxes with separate aspect ratio along with scales to separate the resultant space of bounding boxes different from YOLO which adopts fixed grids. To manage items with numerous measurements, the network merges prediction from numerous feature maps with distinct resolution. Figure 5 demonstrates the architecture of SSD. Provided with VGG16 backbone framework, SSD appends various feature layers with verge of the network. As a result, anticipating the offsets to conventional boxes to variable scales, aspect ratios including related confidence become easier. The network is upskilled with a weighted sum of localization loss as well as confidence loss. Terminating detection solutions are gathered by performing NMS using multi-scale filtered bounding boxes. The SSD 300 runs at 59 FPS and is extra precise as well as well-structured than YOLO, and at the same time SSD is not expert at dealing with objects with smaller size.

DSSD: Deconvolutional single-Shot detector (DSSD) [15] is an enhanced update of single-shot detector (SSD) that appends prediction module as well as deconvolution module and further embrace ResNet-101 as backdrop network. Toward prediction module, a residual block is appended to every predicting layer, and subsequently component-wise inclusion of the results to prediction layer and residual block is done. Deconvolution module increases the resolution of feature maps to strengthen features. Every deconvolution layer accompanied by a prediction module to estimate numerous objects escorted by varied measurement.

Retina Net: In February 2018, Lin et al. [16] introduced focal loss as classification loss function in one-stage object detector, whereas two-stage object detector R-CNN is classical. In the initial phase produces a scanty group of region proposals helping next phase categorize every candidate placement. First stage filters out the majority of negative locations. When compared to one-stage detector, two-stage detector can achieve higher precision which proposes a compact set of candidate locations.

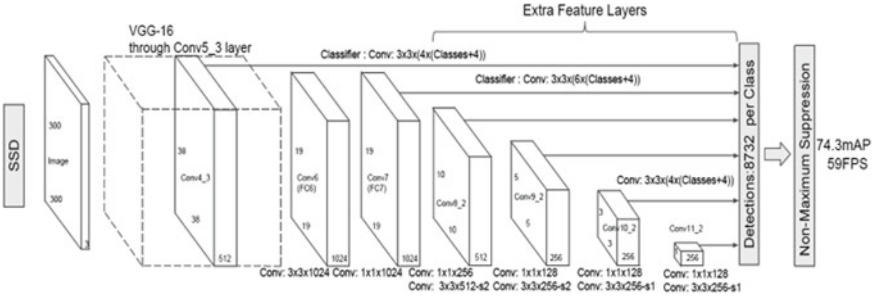


Fig. 5 Architecture of SSD [12], At the end of VGG16 backbone network, SSD appends various feature layers to forecast the offset to conventional anchor boxes including corresponding confidence. Concluding detection outcomes are procured due to performing NMS upon multi-scale refined bounding boxes

The primary reason to this higher precision is utmost forefront–background class imbalance, while one-stage detector trains networks to obtain confluence. Retina Net inherits a rapid speed of preceding one-stage detectors with disadvantages of strenuousness to teach unbalanced positive and negative illustration.

M2Det: M2Det contributes to a huge diversity of scale differences beyond object example, and a multi-level feature pyramid network building additional effectual feature pyramids was proposed by authors [17]. For acquiring final enhanced feature pyramid, authors choose three steps. In the first step like FPN, M2Det adopts multi-level features extraction by several layers at backdrop that acts as the foundation attributes. Next step involves feeding into the block the base feature, comprising changing joint thinned U-shape sections in addition to feature fusion sections, to acquire decoded layers of TUM for feature in succeeding step. Third and final step, a feature pyramid comprising multi-level features is assembled by merging each decoder layers of homogenous scale. Up to aforementioned point features in company of multi-scale as well as multi-level are available. Subsequently rest portion functions to pursue the SSD framework for procuring bounding box localization with classification outcome in a successive mode.

Refine Det: Complete network of Refine Det [18] comprises paired interrelated sections, first the anchors refinement section with second object detection section. Both the sections are attached closed to transfer connection block in order to shift with increasing features out of previous section to improve foretell objects within final section. Each training procedure is an end-to-end method, managed in trio phases, preprocessing, detection (two inter-connected sections), as well as NMS.

SSD, YOLO and Retina Net are classical one-stage detector which uses one-phase regression procedure for acquiring the last output. Furthermore researchers discover allowing usage of two-step cascaded regression procedure will preferably foretell higher quality hard detected objects, particularly those of tiny size objects including furnish extra precise point of objects in image (Table 1).

Table 1 Comparison of various models

Sl. no	Parameters	YOLO	SSD	Retina Net	DSSD	Refine Det	M2Det
1	Model name	You Only Look Once	Single-shot multi-box detector	Retina Net	Deconvolutional single-shot detector	Refine Det	M2Det
2	Backbone network	DarkNet, COSA, DeiT-Ti	VGG MobileNet	ResNet	Feature pyramid network (FPN)	VGG-16 or ResNet-101	VGG-16 or ResNet-101
3	Speed	Low	High	High	High	High	High
	Accuracy (%)	80.3	72.1	83.3 [18]	73.2	81.8	64.6
4	Time	0.84–0.9 sec/frame	0.17–0.23 sec/frame	0.62–0.75 sec/frame	0.20–0.30 sec/frame	0.62–0.75 sec/frame	0.34–0.84 sec/frame
5	Frame per second	45	59	122	50	37	12
6	Mean average precision	0.358	0.251	0.786	0.293	0.29	0.481
7	Number of boxes	~1 k	~8–26 k	~100 k	~8–30 k	~100 k	Default

4 Datasets

Object detection indicates that the object consists in a particular class and locates it in the image. Bounding box is used for localization of an object. Implementing challenging dataset as benchmark is noteworthy in numerous sectors of research, as they sketch a benchmark correlation between various algorithms and place objective for results. Former algorithms concentrate on face detection utilizing numerous ad hoc data records. Subsequently, additional practical and difficult face detection dataset were produced. Other popular obstacle within face detection datasets is to create the dataset. The common object detection datasets, e.g., PASCAL VOC [5], MS COCO [19] and ImageNet-loc [3], are popular guideline for object detection exercise. Furthermore, authorized benchmark is typically assumed to check the functioning of detectors in company of relative dataset. All the fashions to be had at the tensor flow item detection model can be practiced on coco dataset (commonplace objects in context) [20]. This dataset comprise 120,000 photos with a complete 880,000 labeled objects in these images. These models are clever to come across the 90 different varieties of items classified on this dataset. An entire listing of all these different gadgets is available in the statistics part of the skilled model. This listing of items consists of a vehicle, someone, a desk, etc.

5 Applications and Branches

Various areas widely use object detection applications to enable humans to complete tasks, like military domain, security area, transportation area, medical area, etc. We demonstrate the traditional and latest mechanism deployed in this area in detail.

5.1 Safety Field

Face Detection focuses at diagnosing human faces in a photograph. Due to illumination and resolution variation and extreme poses, face detection is still a complex goal. Numerous implementations pay attention on accurate detector designing. To uplift the performance of distinct task, Ranjan et al. [21] discover corresponding work (gender identification, face identification, face landmarks localization and head posture evaluation). To discover unvarying characteristic among near-infrared (NIR) as well as visual (VIS) face images, He et al. [22] instigated a new Wasserstein convolutional neural network technique. Relevant designing of loss functions will improve discriminating ability of DCNNs built on wide-ranging face identification. Researchers in [23, 24, 25, 26] state that the cosine-based SoftMax losses attains considerable victory in deep learning-deployed face identification. Deng et al. [27] propose an additive angular margin loss (ArcFace) toward achieving immensely different attribute aimed at face identification. GUO et al. [16] introduces a fuzzy sparse auto-encoder structure aimed at solo input picture each individual face identification. Readers can consult to [15] for better understanding.

Pedestrian detection emphasizes at identifying walker/pedestrians inside the neural view. Braun et al. [17] emancipate a European City Persons dataset comprising pedestrians, bicyclists and different riders in city busy locations. Real-time walker recognition dedicated to complicacy-conscious cascaded pedestrian spotter.

Anomaly detection assists in swindle identification, weather study, as well as healthcare observance. Present abnormality identification procedures examine the details on a point-wise rationale. To point the expert analyst interesting regions (anomalies) of the data a novel unsupervised procedure termed “Maximally Diver gent Intervals” (MDI) that hunts for adjacent intervals of time including regions in space by Barz et al. [28].

5.2 Defense Area

In defense area, automatic target detection, topographic survey, remote sensing object identification, surveillance mission, flyer identification, intelligent airport security system, anomalous behavior detection, etc., are illustrative applications, which are using object detection mechanism to combat against criminal activity, public safety, terrorism, etc. So, in this kind of application, high-speed computing service molds it to be more impressive, and hence, amid various object detection techniques YOLO is the most appropriate in terms of functioning. For such critical areas, symmetry between network lifetime and object detection techniques is primary concern, and therefore, better approach which focuses on reliability of link becomes important in any wireless network [29].

From a provided video or images identification and classification of military vehicle with assistance of unmanned aerial system is still a challenge and object detection approach can prove to be beneficial in this aspect. During battle automatic target detection in battlefield from captured images with the help of object detection can be helpful.

6 Results

A system's or algorithm's result analysis is based on a set of parameters. Performance, time spent, resources required, accuracy and other factors are commonly used in practically all analyses. The performance is a parameter that indicates how well the algorithm works. The time consumed to evaluate the algorithm and obtain the output is represented by the parameter time taken. The quantity of resources required by the algorithm is defined as resources needed. Accuracy is the algorithm's promising factor, defined as the percentage of accurate output produced by the algorithm.

When the common conditions are applied to Ross Girshick's R-CNN techniques of object detection, the outcome reveals that R-CNN is significantly speedy than the conventional techniques based on classification techniques [30, 31]. RCNN uses a selective search to extract only 2000 regions per image, rather than a large number of regions. As a result, the feature extraction will only cover 2000 regions. R-CNN still has its limitations after such a substantial reduction in computation. To begin with, training the network takes a long time because each image is classified into 2000 regions. Second, it is not applicable in real time because each test image takes around 40 s to process. Finally, because it employs a preset selective search strategy, it is unable to learn from past experiences.

Ross Girshick later designed a latest type of R-CNN entitled fast R-CNN to address the R-CNN's flaw. Unlike R-CNN implementation the input picture is supplied to the CNN instead of the region proposals in this method. To locate image's proposal region, CNN creates a convolutional feature map. For each object, a feature vector is produced from the feature map, and the softmax layer is utilized to forecast the class of the proposed area based on this vector increasing the effectiveness of this technique.

It is not mandatory to provide CNN 2000 area proposals every single time, which is considerably superior to R-CNN. Instead, each image is subjected to a single CNN process. Another method, alike R-CNN including fast R-CNN, is suggested. The approach is implemented same way as the former techniques, but rather than selective search algorithm, an independent network is utilized to forecast the proposed regions. The ROI polling layer is used to reshape the proposed regions, which are subsequently utilized to separate and identify the classes and borders. Because an independent network rather than a fixed technique is utilized to anticipate the proposed region, this technique is substantially speedy than fast R-CNN.

For object recognition, a recent method YOLO is presented. Because previous methods rely on proposed areas to recognize the object in the image and never whole image is considered. Object detection is performed on regions with a high possibility of containing items. However, in YOLO, there is simply one convolutional network, which analyzes the entire image. The image is divided into a S × S grid, accompanied by m bounding boxes. The network produces a class probability for each box, including the classes with greater probability compared to the threshold value which are utilized to find the object. Due to its single convolutional neural network, this technique has numerous advantages. The whole image is assessed once, and bounding boxes and class probability are calculated. Second, the whole detection procedure is conducted in a single network, making network optimization simple. Because it just has one convolutional neural network, it is substantially speedy than the R-CNN, fast R-CNN and faster R-CNN.

Authors in paper [32] compares various models in terms of delay, mean average precision (mAP), frames per second (FPS) and usage in real-time applications. Aforementioned paper ahead without doubt illustrates that YOLO outperforms R-CNN-based algorithms in terms of latency and frame per second (FPS). It is evident that a precision trade-off was made in order to attain this speed. Despite its small mean average precision (mAP), YOLO has an acceptable for real-time applications, and once combined with its high FPS and latency, it is evident that it is the finest algorithm in its class.

7 Conclusion and Trends

Deep learning-centered object detection technology has matured swiftly along with increase the regular upgrade of strong and intensive computing equipment. To implement more dynamically and precise applications, there is a requirement of a high-precision real-time system. Researchers have developed various directions to attaining highly efficient precision detectors like building a current architecture, squeezing rich features, utilizing good representation, upgrading processing speed, train from beginning, techniques without anchors and troubleshooting complex 40 to 4 scenes such as small objects and occluded objects. Implementation of object detection is slowly expanded in the security area, military area, transportation area, medical area and life field with the increasingly influential object detectors. In recent days, but there is still a lot to achieve in future development.

Due to the importance of speed and accuracy in object detection applications, it is crucial to maintain a small computation time and to process the input speedily in order to give the user with the output. Because it only has one neural network and works in real time, YOLO is the ideal option for real-time object detection. Quick computation time. It can produce results faster than other approaches, and the precision of the technique can be handled according to the system's needs.

7.1 Trends

Hybrid stage detector: Hybrid stage detector is combination of various stage detectors like one-stage detector and two-stage detector. Such detector helps researcher gain higher precision and maximum throughput in real-time applications.

Video object detection: *Detection* of object in real-time video, operational videos are important research areas and are extremely challenging due to problems like motion blur challenging to detect object in moving video, hazy motion, defocus of video, motion target vagueness, extreme target movement, tiny targets, obstruction and abridged. To achieve a better performance is really a difficult task.

Multi-domain object detection: High detection interpretation regularly requires domain-specific detection on the specified dataset. Therefore, a general detector which is efficient on functioning on numerous multi-domain detector, image domains can resolve this issue without previous understanding of current discipline. Domain shift is a bothering assignment for future research.

3-Dimension object detection: When juxtaposed to 2D image-based detection, 3-D object detection has become demanding research orientation with the innovation of 3-D sensor and manifold application of 3D knowledge. "Light Detection and Ranging" (LiDAR) point cloud produces genuine depth details which can be utilized to precisely discover objects and personify their forms. In 3-D space, LiDAR also permits correct localization of objects.

Salient Object Detection: Salient object detection (SOD) has a difficulty in spotlighting major object regions in pictures while in video object identification classification as well as location of objects of interest in a continual arena is done. SOD is executed in a vast scope of object-level application in so many disciplines. Precise "object detection" in videos is offered by SOD by extracting object regions of interest in every frame. Hence, for prominent identification and difficult detection mission, spotlighting target detection is a pivotal preparatory procedure.

Multi-source information assistance: Nowadays, with the growth of big data techniques including widespread usage of Internet community, multi-source details are becoming effortless to approach. Numerous social media details can supply both images and videos, and their representation in textual form can assist identification duty. Blending multi-source data is becoming apparent study inclination with the emerging of numerous techniques.

Medical diagnosis and imaging: With the encouragement, artificial intelligence deployed medical gadget in April 2018, Food and Drug Administration (FDA) in US initially recommended a diabetic retinopathy identifier having correctness of 87.4% which was based on artificial intelligence software named IDx-DR. Amalgamation of image identification systems as well as mobile devices can prepare cell phone as strong family investigative device for anybody. Aforementioned inclination is packed with obstacles at the same time with too much hopefulness.

Real-time detection and remote sensing airborne: Precise investigation of remote sensing image, automated detection software and integrated hardware are essential in both agriculture fields and military services, and this will escort unmatched development in this area. System on chip (SoC) realizes real-time high-altitude detection while supplying to deep learning-focused object detection methodology.

Advanced medical biometrics: Researchers started studying utilization of deep neural network, by using neural networks to recognize the possibility of heart disease by inspecting the retinal images and speech pattern. Medical biometrics and its application will be used for passive monitoring in near future.

References

1. Khan A, Sohail A, Zahoora U, Qureshi AS (2019) A survey of the recent architectures of deep convolutional neural networks. arXiv preprint arXiv:1901.06032
2. Zou Z, Shi Z, Guo Y, Ye J (2019) Object detection in 20 years: A survey. CoRR, abs/1905.05055
3. Liu L, Ouyang W, Wang X, Fieguth P, Chen J, Liu X, Pietikainen M (2018) Deep learning for generic object detection: A survey. arXiv preprint arXiv:1809.02165
4. Jiang H, Learned-Miller E (2017) Face detection with the Faster R-CNN. In: 12th International Conference on Automatic Face & Gesture Recognition. IEEE, Washington DC USA
5. Yang Z, Nevatia R (2016) A multi-scale cascade fully convolutional network face detector. arXiv:1609.03536v1[cs.cv]
6. Lin T, Dollar P, Grilshick R, He K, Hariharan B, Belongie S (2017) Feature pyramid network for object detection. In: Conference on Computer Vision and Pattern Recognition (CVPR), 936–944. IEEE, Hawaii
7. Girshick R, Donahue J, Darrell T,Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: 14th International Conference on Computer Vision, IEEE, Ohio
8. Redmon J, Divvala S, Girshick R, Farhadi A (2016) You only look once: Unified, real-time object detection. In: International Conference on Computer Vision and Pattern Recognition. IEEE, Las Vegas
9. Redmon J, Farhadi A (2016) YOLO9000: better, faster, stronger, 7263–7271. arXiv:1612.08242
10. Li Y, Lu Y, Che J (2021) A deep learning approach for real-time rebar counting on the construction site based on YOLOv3 detector. Automation Const 124:1–14
11. Liu Z, Li J, Shu Y, Zhang D (2018) Detection and recognition of security object based on Yolo9000. ICSAI. IEEE, Nanjing
12. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C-Y, Berg AC (2016) SSD: Single shot multi-box detector. In: European Conference on Computer Vision. Springer, Cham
13. Erhan D, Szegedy C, Toshev A, Anguelov D (2014) Scalable object detection using deep neural networks. In: International Conference on Computer Vision, IEEE, Ohio

14. Bell S, Lawrence Zitnick C, Bala K, Girshick R (2016) Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks. In: CVPR, IEEE. Las Vegas
15. Wang M, Deng W (2018) Deep face recognition: a survey. https://arxiv.org/abs/1804.06655
16. Guo Y, Jiao L, Wang S, Wang S, Liu F (2017) Fuzzy sparse autoencoder framework for single image per person face recognition. IEEE Trans Cybernatics 48(8):2402–2415
17. Li X, Flohr F, Yang Y, Xiong H, Braun M, Pan S, Li K, Gavrila DM (2016) A new benchmark for vision-based cyclist detection. In: Proc IEEE Intell Vehicles Symp (IV), pp 1028–1033. IEEE, Sweden
18. Tetila EC, et al (2020) Detection and classification of soybean pests using deep learning with UAV images. Comp Elect Agri 179:1–11
19. Chen C, Seff A, Kornhauser AL, Xiao J (2015) Deep driving: learning affordance for direct perception in autonomous driving. In: 15th International Conference on Computer Vision, IEEE, Chile
20. https://cocodataset.org
21. Ranjan R, Patel VM, Chellappa R (2019) 'HyperFace: a deep multitask learning framework for face detection, landmark localization, pose estimation, and gender recognition. IEEE Trans Pattern Anal Mach Intelligence 41(1):121–135
22. He R, Wu X, Sun Z, Tan T (2019) Wasserstein CNN: learning invariant features for NIR-VIS face recognition. IEEE Trans Pattern Anal Mach Intelligence 41(7):1761–1773
23. Zhang X, Zhao R, Qiao Y, Wang X, Li H (2019) AdaCos: Adaptively scaling cosine logits for effectively learning deep face representations. arXiv:1905, 00292
24. Liu Y, Li H, Wang X (2017) Rethinking feature discrimination and polymerization for large-scale recognition. arXiv:1710.00870
25. Ranjan R, Castillo CD, Chellappa R (2017) L2-constrained softmax loss for discriminative face verification. arXiv:1703.09507. https://arxiv.org/abs/1703.09507
26. F. Wang, X. Xiang, J. Cheng, and A. L. Yuille.: NormFace: L2 hypersphere embedding for face verification. In: Proc. 25th ACM Int. Conf. Multimedia, ACM, 1041–1049 (2017).
27. Deng J, Guo J, Xue N, Zafeiriou S (2018) ArcFace: additive angular margin loss for deep face recognition. arXiv:1801.07698. Available: https://arxiv.org/abs/1801.07698
28. Bjorn B et al (2018) Detecting regions of maxima divergence for spatio-temporal anomaly detection. IEEE Trans Pattern Anal Mach Intell 41(5):1088–1101
29. Babulal KS, Tewari RR (2011) Cross layer design with link and reliability analysis for wireless sensor network. In: Proceedings of 2nd International Conference on Current Trends in Technology, IEEE. Nirma University Ahmedabad
30. Khare M, Thanh Binh N, Srivastava RK (2014) Human object classification using dual tree complex wavelet transform and Zernike moment. In: Transaction on large scale data and knowledge centered system XVI, LNCS, 87 101
31. Kumar P, Thakur RS (2021) An approach using fuzzy sets and boosting techniques to predict liver disease. CMC-Computers Mat Cont 68(3):3513–3529
32. Kumar P et al (2019) A comparative study of object detection algorithm in a scene. Int J Eng Res Tech 8(5):1–3

Bi-Modal Meta-Classification of Tweet Spamicity Using Machine Learning Approach

P. Jayashree, K. Laila, V. Vinuvarsidh, and K. Santhosh Kumar

Abstract Nowadays, social media plays a centric role in sharing and disseminating information on products and services. Social network users post messages for attracting followers and increasing popularity. This attracts spammers to post fake and untruthful opinions on products. So, a strong and dependable framework for distinguishing spam users is important. Most of the research on detecting spam focuses on either tweets or metadata from Twitter profiles. Generally, spam tweets are generated using automated tools. The existence and credibility of the users in the social network is an important parameter in deciding the tweet is spam or not. In this work, a meta-classification approach is used with dynamic features extracted from the tweets, and tweeter behavior. The proposed work confirms spamicity in tweets using an aggregated model with spamicity inference generated by LSTM and machine learning models. Machine learning models are selected as it learns to extract patterns and interpret results from large dataset efficiently. The identification of distinct behavioral patterns from tweets is potent in various practical applications such as automatic knowledge extraction fields and large-scale human content generated platforms. The performance of the bi-modal meta-classifier is tested with benchmark datasets, and the accuracy of 99.76% achieved is found to be better than some recent related works.

Keywords Naïve Bayes · Random forest · Spam · Tweets · Twitter · LSTM · Bot

1 Introduction

Twitter is a social networking and blogging service developed in 2006 [1]. Users can post information using a functionality called tweets and can post links. Spammers use this facility to post harmful URLs which may contain malware downloads, scams, phishing. The tweets are shared in the timeline of the user. Tweets also use "hashtags" to highlight the topic of the messages. The growing attractiveness of Twitter and

P. Jayashree · K. Laila (✉) · V. Vinuvarsidh · K. S. Kumar
Department of Computer Technology, MIT Campus, Anna University, Chennai, India
e-mail: lailagodson1@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_51

the fact that there are 199 million active users on Twitter make this platform a tempting option and forum for spammers to spread misinformation and to promote their products. The problem with current spam filters is that they only use information extracted from the tweets to detect whether the tweets are spam or not and this method will not work if the spammer finds a way to write a tweet that could not be identified by the spam filter tools like Google safe browsing. Social networking experts say that almost 30% of social networking accounts are spam accounts [2]. So, spam detection in social networks is an important step in identifying a spammer. Instead of just analyzing the tweets and making a decision based on content analysis, the proposed method analyzes the spammer's behavior on Twitter by processing his tweets posted contents and also takes various other parameters like no. of URL's, no. of hashtag, mean, and variance of inter tweet delay (timestamps), no. of times the user has been mentioned, and account age into consideration. The entire process is organized into three sub-models including content model, user model, and model aggregation, each analyzing various aspects of the suspected user. These in-depth analyzes and experiments help us better understand the behavior of the user as it forms a pattern which better depicts the spammer's profile. However, an automatic account for propagating spam activities is still a demanding task. Employing LSTM on sequence tweets that preserve the text continuity will help in identifying the spam words more correctly makes our paper more elaborate for the prediction. Machine learning-based classification techniques have the ability to handling overfitting issues on short and noisy text effectively.

Main contributions are as follows:

- We proposed a meta-classification model based on bi-modal data for detecting spam.
- Dynamism in tweet and tweeter behavior is considered a prominent feature for spam detection.

2 Related Works

2.1 Spam Detection on Twitter

Spam identification in social networks is a moderately late zone of research. In [3], it was proposed that spam tweets usually contain at least one malicious URL. The collected tweets were filtered to examine the potential spamicity and the features were extracted to identify the spam clusters in real time. In [4], it was stated that tweets are easy to classify just by taking the context-based text representation on the social platform which also justifies the statement proposed in [5]. The empirical study of the model highly relied on textual content with binary classifiers. Some important parameters include number of initial URL's, tweet length, inter-tweet duration, and message the tweet implies [6] to address the challenges like data sparsity, propagating misinformation, and scalability. The number of URLs in a tweet is a major parameter in determining the trustworthiness of the user.

2.2 Spam Detection Methods

There are many methods to identify spam tweets on Twitter. In [7], the authors proposed that one way is to keep track of all the spam words and calculate the percentage of spam words found in a tweet. Another way is to check the relevance between the actual tweet and the hashtags used. It is often time-consuming and inaccurate in some cases. In [8] and [17], the authors showed the importance of the trustworthiness of the entities involved in successful communication as well as identifying attackers. In [9], an adaptive classification method was proposed for spam detection in Twitter. This method works by detecting the spam words list and URL to predict the spam. The encoder-decoder approach was introduced for vectorization and text summarization and predicted with 50% loss for similarity score prediction. The authors of the paper [10], proposed a spam classification method based on credibility which is determined using two types of features: content-related features and user-related features. Fake news detection integrated with opinion mining on Arabic tweets produced 76% accuracy, and it demands feature engineering for better inference. The credibility of the tweets is predicted using the tweets and the user credibility is determined by the features as mentioned in [11, 12].

Social media users often prefer ungrammatical language, misspelled words, and non-standard short forms for their posts. This becomes a challenge for the content-based analysis techniques. It is necessary to convert them to canonical form. In [13, 14] the authors proposed Peter Norwig's algorithm to normalize the contents. It considered non-standard words as noises. The noisy zone recognition method is used to identify the areas of interest of the content using a dictionary and the canonical form is recognized using the correct zone recognition method. It is discussed that the GRU-based lexical normalization algorithm [15] examines the excellency of the embedding models on character and word-level normalization. The authors also found that pre-trained ELMo outperforms BERT embedding due to the ability to capture contextual representation. In [16], it was discussed that in Twitter the spammers are detected based on their ability to detect fake content, URL content, and trending topics. The content polluters on social networking platforms create fake trends.

In [18], sentiment analysis was used to identify the pattern of the tweets and then classified using a pattern-based classifier. The suspicious activity of the Twitter account was handled not only based on patterns but also by modeling a set of features extracted out of consumption of social account and users' opinions. In [19], the authors proposed the credibility analysis method to find the credibility of the information in the tweets and determined the reputation of the user using naıve Bayes, random forest, and a feature ranking algorithm. The authors in [20] examined how spammers used the hashtags in Twitter to manipulate the platform. In [21] the authors proposed a partially supervised learning-based spammer group detection (PSGD) algorithm to use semi-supervised learning models. In this paper, the authors labeled some spammer groups and classified the unlabeled spammer groups using the naıve Bayes model, and expectation–maximization (EM) algorithm is used to improve the NB model. In [22], authors examined the effects of feature selection technique in online review sources which are also upheld by [23] to predict the opinion credibility of social network users. Madisetty, et al. proposed an ensemble approach [24] to achieve better accuracy by integrating pre-trained embedding techniques with CNN and feature-based classification models.

2.3 Bot Detection

In [25], the authors proposed that the users on social media can be classified into active spammers, advertisers, genuine users, and automated accounts. In online social networks (OSN) like Twitter, the contribution of the automatic bot users (autobots) exceeds normal users. These autobots are necessary to be eliminated for better and fair predictive analysis. The statistic enhancers are considered as bots as they artificially increase the reputation of the end-user or organization. In [26], the authors proposed that the characteristics of the autobots can be predicted using the parameters like reputation, profile completeness, length of user names, account age, tweet patterns, tweeting interval, content of the tweet, tweet similarity, etc. It is also discussed that a pattern for the autobots is recognized using these features that help to predict the autobot users. The naive Bayes algorithm is used to analyze the patterns and predict the spammers. In [28], the authors proposed machine learning algorithms such as decision trees, logistic regression, and support vector machine to predict the autobot users by analyzing the above-stated features.

Based on the literatures surveyed, some characteristics of spammers are observed as follows. Spammers in general are greedy in increasing the followers_count and more active in wide conversation with less relevant information. They are found to be more dynamic with continuous tweets and retweets. So, it is proposed to find out the dynamism from tweets and twitter account data using meta-classification techniques. Data volatility issues can be addressed with sequential model by taking the sequential formation of the tweet data. Alleviating bot activity on the social platform using modern sophisticated approaches becomes increasingly obvious. The proposed work considered non-bot records of the accounts for better generalization capability.

3 Proposed Work

A meta-classification architecture combining bi-modal classifiers for spamicity analysis in tweets is devised as shown in Fig. 1. The framework comprises three models, namely, content model, user model, and model aggregation.

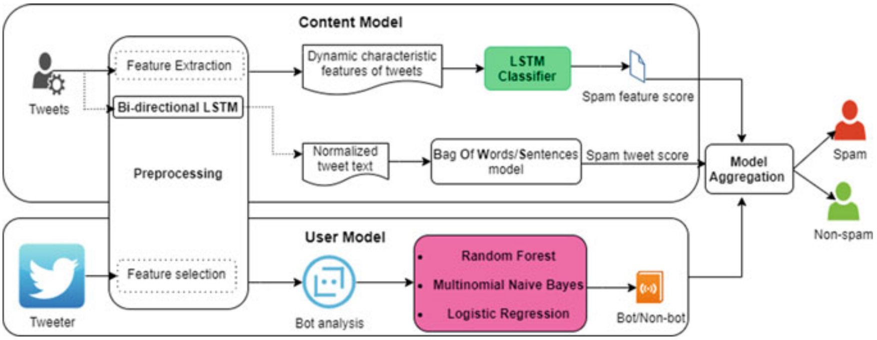


Fig. 1 Framework for the bi-modal spam classification

3.1 Content Model

The content model uses the tweet as the modality and employs deep learning to evaluate the tweet spamicity. The tweet text and other dynamic characteristics features of the tweets like percentage of tweets posting duration, mean, and standard deviation of inter-tweet delay are explored towards better spam detection. The text and tweet characteristic features are preprocessed to extract the required data for further classification as explained below.

Pre-processing. It helps to decide the granularity of the text by using a tokenizer that splits the documents into tokens based upon special characters and white space. This tokenizer can extract more information than commonly used word analyzers. The tweet dataset is subjected to some pre-processing techniques. ML models can understand only vector of numbers, so the text data has to be converted to a vector of numbers. These techniques include tokenization, stop words removal, and stemming. Tokenization is the process of splitting the sentence into individual words, whereas stop words removal is the process of removing unwanted words, numeric, and null values. Stop words include—is, was, has, have, etc. Finally, words undergo the process of stemming, which converts words to its base form, for example, moving will be converted to move. NLP packages are used for performing these operations ML models cannot work directly with the text. It has to be pre-processed and converted into vector of numbers.

Feature Extraction. The tweets of the user are not alone sufficient to predict the characteristics of the user. Effective spam filtering depends upon how well the user is examined in the social network. The dynamic features of the user's Twitter account are extracted using the Twitter API. It includes description, verified, age of the user account (in days), number of following and followers, reputation, user mention (@) ratio, unique user mention (@) ratio, URL ratio, hashtag (#) ratio, an average of tweet content similarity, retweet and reply rate, number of tweets per day/week, mean and standard deviation of inter-tweet delay, and number of tweets posted by the user at various time intervals of the day.

Text Normalization. The normalization of the text here represents the conversion of incomplete words to a completed form. Existing normalization algorithms work by either finding the smallest distance between two words or making use of phonetics to normalize. Instead of these methods, our method makes use of bi-directional LSTM network to memorize the position of words in a sentence for better accuracy. This technique is applied to the tweets.

Classification. The continuous stream of tweets is analyzed using the LSTM model which works best for time series data. This approach is explicitly designed to avoid the long-term dependency problem and it is capable of learning prominent features from the training data and storing information over an extended period of time. Unlike general neural networks where processing is independent of the previous layers, LSTM helps to memorize and pass the error information to the next layer.

Spam Score Calculation. The spam score on tweets is calculated based on the presence of spam words and spam sentences over a threshold. Consider the tweet “click this link to get the phone”, “Limited period only”. The bag of words corresponds to [‘click’,’link’,’get’,’offer’,’phone’,’limited’,’only’,’period’]. Here “click, link, offer” are spam words, whereas “limited period only”, “you will not believe it”, “directly from company internet marketing” are considered as spam sentences. The process for generating spam scores for tweets is defined in Algorithm 1.

Algorithm 1 Spam Score Calculation Spam Tweet.

```
Input: Tweets of users
Output: Percentage of Tweets that are spam
B1 ← spam_words
B2 ← spam_sentences
spam score ← 0
for every tweet T of user do
    Tokenization and stemmatization [tweet_words]
    for word in T do
        if word in B1 then
            spam score+ = 1
        end if
    end for
    for sentence in T do
        if sentence in B2 then
            spam score+ = 1
        end if
    end for
end for
```

Feature Statistics. The present work explores the dynamic characteristic features to distinguish spam and benign tweets. The cumulative distribution functions (CDF) of dynamic features have been analyzed which discloses their discriminative power between spammers and non-spammers as plotted in Fig. 2 for Twitter dataset. As highlighted in Fig. 2a, the reputation of the user; perhaps the clearest indicator of the credibility of a user is its number of followings and followers. Non-spammers generally have high followers, and a low number of followings list contrary to spammers in Fig. 2b, c. In Fig. 2d, spammers generally face difficulty in generating lengthy text. Number of tweets per day and day-long sessions without any short break increases the probability of that account being a spammer shown in Fig. 2e. However, lower mean and standard deviation intertweet_delay indicates a higher probability of spammers. Like in Fig. 2f spammers create new accounts very frequently and have less account lifetime. As in Figs. 2g–i, spammers have a higher number of advertisement URLs, retweets, and mentions, respectively.

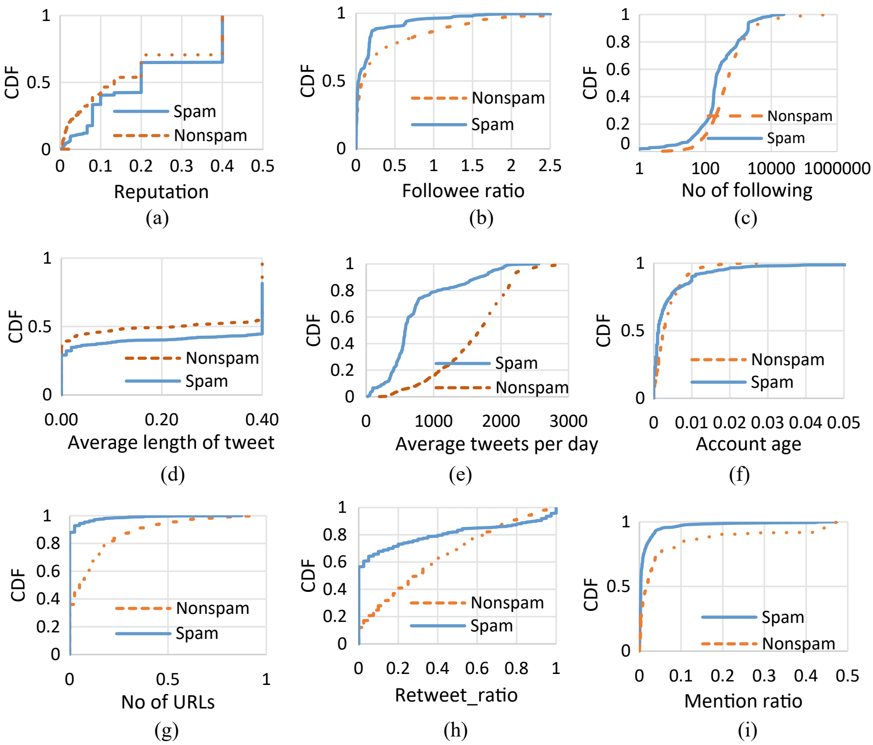


Fig. 2 Cumulative distribution functions of dynamic characteristics features

3.2 User Model

The main goal of the user model is to analyze the behavior of bot or non-bot user account in Twitter by extracting and experimenting with various features like description, retweet ratio, unique mentions, screen_name, no. of tweets per week, listed_count, URL, created at, favorite_count, verified, status_count, followers to the following count, reputation, etc.

Feature Selection. Pearson correlation coefficient is used to measure the correlation or the linear relationship of various features. It drops the features which have a lower correlation score for the target class by setting the threshold of 0.5.

Bot Analysis. The Twitter bot is a software program that uses Twitter API to control the Twitter account. Bots can autonomously perform operations like a retweet, tweet, follow, or unfollow other users or send spam messages to other users. The "follower count" parameter is used in finding many other parameters like retweet and reputation. So, there is a need to optimize that parameter in order to maximize accuracy as much as possible. The list of all followers of the suspected user and their behavior is analyzed by extracting relevant parameters like number of tweets posted by the user in different intervals, reputation, followers and following, URL ratio, content similarity, and posts liked by him, and frequency at which he likes tweets and retweets it. It also exploits bag of words containing words that are most commonly used by automated bots in description, account name, location, and URL section to decide the probability of these accounts being a bot.

Classification. Models like gaussian naive Bayes (GNB), logistic regression (LR), and random forest (RF) are used for bot classification. These supervised learning techniques are widely used for various detection tasks. Logistic regression uses logistic or sigmoid function that maps the probabilities from predictions. Gaussian naïve Bayes based on bayes theorem predicts the probability of before and after getting the evidence. Random forest is a collection of decision trees which preclude overfitting issues.

3.3 Model Aggregation

The efficiency of proposed bi-modal model combining the classification using tweet and tweeter characteristics is analyzed using a weighted aggregation. Weights are assigned for the outputs obtained from tweet and tweeter characteristics based on the importance of the entities. This weighted average is enhanced with the user characteristics for better prediction of spam tweets.

For finding the weighted average, the average spam score is considered as a parameter for defining weights. The average spam feature score (SF_S) is 0.660374. The average spam tweet score (ST_S) is 0.432429. Accordingly, weights are assigned as $W_1 = 0.6$ and $W_2 = 0.4$. The final score is calculated as follows.

$$\text{Final spam score} = W_1 * \mathrm{SF}_S + W_2 * \mathrm{ST}_S \tag{1}$$

The bot detection is applied to eliminate malicious users. For non-bot users based aggregated spam score thresholding is applied to conclude as spam or not.

4 Experimentation

4.1 Dataset Description

This work employs three different datasets to analyze tweet and tweeter spam classification.

Tweet Dataset. Tweet dataset with the total number of 5573 tweets is considered, where 4854 are ham and 749 tweets are spam. In addition, 23 tweet features of 1331 users have been extracted in real-time using Twitter API and manually annotated.

Statistical Dataset. The statistical dataset contains 30 features on statistical information of Twitter users is used.

Bot Dataset. A dataset containing information of 1057 bots and 1177 non-bots is obtained from Kaggle. Features in the dataset include description, verified, age, screen name, location, URL, followers count, following count, listed count, created at, lang, status, default profile image, has extended profile image, name, and a label denoting whether the user is bot.

4.2 Experimental Setup

The experimental model implements the proposed model on Ubuntu 16.04 64-bit machine with a 3 GHz Intel CPU and 16 GB memory. To implement the deep learning model, the experimental framework employs the required python libraries specifically Tensorflow and Keras. With the assistance of the python version 3.6.8 in spyder 5.1.5, machine learning libraries, NLTK pre-processing, and SK-learn libraries for the tweet and bot processing are performed.

4.3 Classification and Evaluation

The text is normalized using bi-directional LSTM. The pre-processed tokenized words are normalized using the classifier which makes use of three encoding layers and two decoding layers to memorize position words in a sentence. This is because the word "mt" may mean "empty" in one sentence, and "mount" in another. This classifier makes it possible to normalize words based on their occurrence in the sentence. Tweet representation using word2vec with Gensim, an NLP toolkit for vector space, and topic modeling is used to learn new vectors from tweets. The size of the vocabulary is 213,301 each with 300 dimensions. This sequential model is built on hyper-parameters of embedding which maps the input word into a 300-dimensional vector. Model compilation done using *rmsprop* optimizer and binary cross_entropy is trained as the loss function for classification. The model is fit to the training set with a batch size of 64 and 10 epochs and a training–testing split of 80–20%.

LSTM is employed for classifying tweets based on 23 features. The input vector consists of 23 features and the output has two elements corresponding to spam or not spam. The LSTM model is applied in the hidden layer. We employed the sigmoid activation function for the binary classification to evaluate the content model. The performance of the content model is evaluated using accuracy, precision, recall, F1-score, and mean absolute error (MAE). Which is defined in the following equations.

User model is evaluated using classical machine learning classifiers like gaussian naive Bayes (GNB), logistic regression (LR), and random forest (RF) to identify bots and the performance is analyzed using accuracy, precision, recall, and F1-score.

- **Precision**: It measures the percentage of truly positive out of all the predicted positive values.

$$\frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \tag{2}$$

- **Recall** It measures the percentage of predicted positive out of the total number of positive.

$$\frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \tag{3}$$

- **F1-score**: It captures the classifiers' prediction quality by requiring higher score of the harmonic *mean* of both recall and precision.

$$\frac{2 * (\text{Precision} * \text{Recall})}{\text{Precision} + \text{Recall}} \tag{4}$$

- **Accuracy**: It is the ratio of total number of correctly classified instance to the total instances.

$$\frac{\text{True Positive} + \text{True Negative}}{\text{True Positive} + \text{True Negative} + \text{False Positive} + \text{False Negative}} \tag{5}$$

- **Mean Absolute Error**: It takes the average squared difference between outputs (y_i) and targets $\widehat{y_i}$. N implies the total number of data instances.

$$\mathrm{MAE} = \frac{1}{N}\sum_{i}^{N} |y_i - \widehat{y}_i| \tag{6}$$

5 Results and Discussion

5.1 Performance of the Content Model

The spam detection performance is analyzed on individual models and compared with the aggregated model. The sample set of predicted spam features and tweet scores is enclosed in Fig. 3a and b. The spam detection efficiency is calculated from the spam feature score by applying a threshold of 0.5. The efficiency of 98.4% is achieved as inferred from Fig. 4a, and MAE is observed from Fig. 4b.

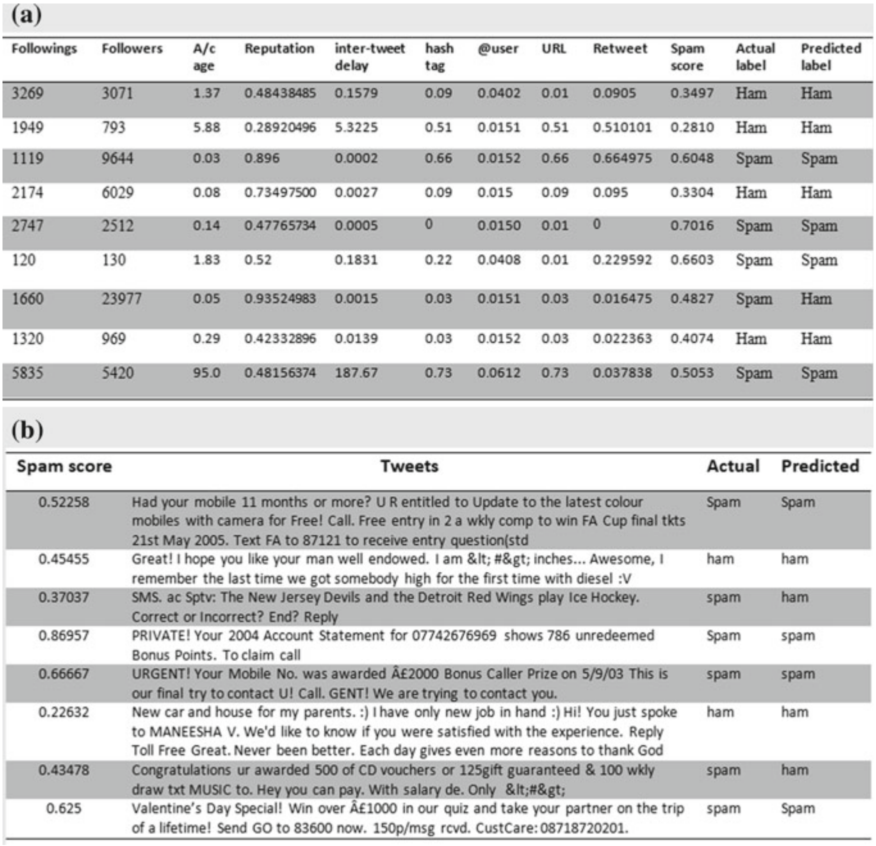
5.2 Performance of the User Model

Bot detection efficiency of the user model is evaluated using classifiers gaussian naive Bayes, logistic regression, and random forest. From the results, it is found that random forest outperforms other classifiers with an accuracy of 97%. Since it is capable of preventing overfitting and gives a generalizable outcome and relatively high accuracy. It employed 100 trees and Gini impurity as splitting criteria. Related performance metrics are shown in Fig. 5.

Figure 6a provides information on the behavior of the bot with respect to followers and following. It is found that bots or active users have many followers although bots don't have a lot of followers, they follow other users just as equally as non-bot users.

The graph in Fig. 6b illustrates that the bots have an uneven distribution of following vs followers, whereas non-bots have an even distribution of followers' vs following ratio, indicating that the bots randomly send friend requests to various users. ROC curves for these models on bot detection are plotted in Fig. 7.

The proposed model is compared with neural network-based ensemble methods [24] and LSTM with attention model [27] as tabulated in Table 1 and the supremacy of the predictive performance of the proposed model in identifying spam tweets is as the proposed model extracts the sequential context from tweets toward computing spamicity on non-bot records.



(a)

Followings	Followers	A/c age	Reputation	inter-tweet delay	hash tag	@user	URL	Retweet	Spam score	Actual label	Predicted label
3269	3071	1.37	0.48438485	0.1579	0.09	0.0402	0.01	0.0905	0.3497	Ham	Ham
1949	793	5.88	0.28920496	5.3225	0.51	0.0151	0.51	0.510101	0.2810	Ham	Ham
1119	9644	0.03	0.896	0.0002	0.66	0.0152	0.66	0.664975	0.6048	Spam	Spam
2174	6029	0.08	0.73497500	0.0027	0.09	0.015	0.09	0.095	0.3304	Ham	Ham
2747	2512	0.14	0.47765734	0.0005	0	0.0150	0.01	0	0.7016	Spam	Spam
120	130	1.83	0.52	0.1831	0.22	0.0408	0.01	0.229592	0.6603	Spam	Spam
1660	23977	0.05	0.93524983	0.0015	0.03	0.0151	0.03	0.016475	0.4827	Spam	Ham
1320	969	0.29	0.42332896	0.0139	0.03	0.0152	0.03	0.022363	0.4074	Ham	Ham
5835	5420	95.0	0.48156374	187.67	0.73	0.0612	0.73	0.037838	0.5053	Spam	Spam

(b)

Spam score	Tweets	Actual	Predicted
0.52258	Had your mobile 11 months or more? U R entitled to Update to the latest colour mobiles with camera for Free! Call. Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive entry question(std	Spam	Spam
0.45455	Great! I hope you like your man well endowed. I am &lt; #&gt; inches... Awesome, I remember the last time we got somebody high for the first time with diesel :V	ham	ham
0.37037	SMS. ac Sptv: The New Jersey Devils and the Detroit Red Wings play Ice Hockey. Correct or Incorrect? End? Reply	spam	ham
0.86957	PRIVATE! Your 2004 Account Statement for 07742676969 shows 786 unredeemed Bonus Points. To claim call	Spam	spam
0.66667	URGENT! Your Mobile No. was awarded Â£2000 Bonus Caller Prize on 5/9/03 This is our final try to contact U! Call. GENT! We are trying to contact you.	spam	spam
0.22632	New car and house for my parents. :) I have only new job in hand :) Hi! You just spoke to MANEESHA V. We'd like to know if you were satisfied with the experience. Reply Toll Free Great. Never been better. Each day gives even more reasons to thank God	ham	ham
0.43478	Congratulations ur awarded 500 of CD vouchers or 125gift guaranteed & 100 wkly draw txt MUSIC to. Hey you can pay. With salary de. Only &lt;#&gt;	spam	ham
0.625	Valentine's Day Special! Win over Â£1000 in our quiz and take your partner on the trip of a lifetime! Send GO to 83600 now. 150p/msg rcvd. CustCare: 08718720201.	spam	Spam

Fig. 3 **a** Sample spam feature score **b** sample spam tweet score

6 Conclusion

Extracting useful information and identifying the veracity of the content from a complex network is a challenging task. The proposed work focused on extracting dynamic characteristic features of the tweets and tweeter account for spam detection. The proposed meta-classification model employed deep learning LSTM and other machine learning models. The proposed work is focused on various data modalities. The proposed model is validated with a benchmark dataset and is evaluated to achieve better accuracy of 99.76% compared to some other related works. The proposed model reduces type I and type II error effectively. This work can be used as a base for potential applications like chatbot and instant messaging bot applications and in various domains including healthcare. In the future, multimedia-based linguistic features and sentiment features can be included to build a still better model.

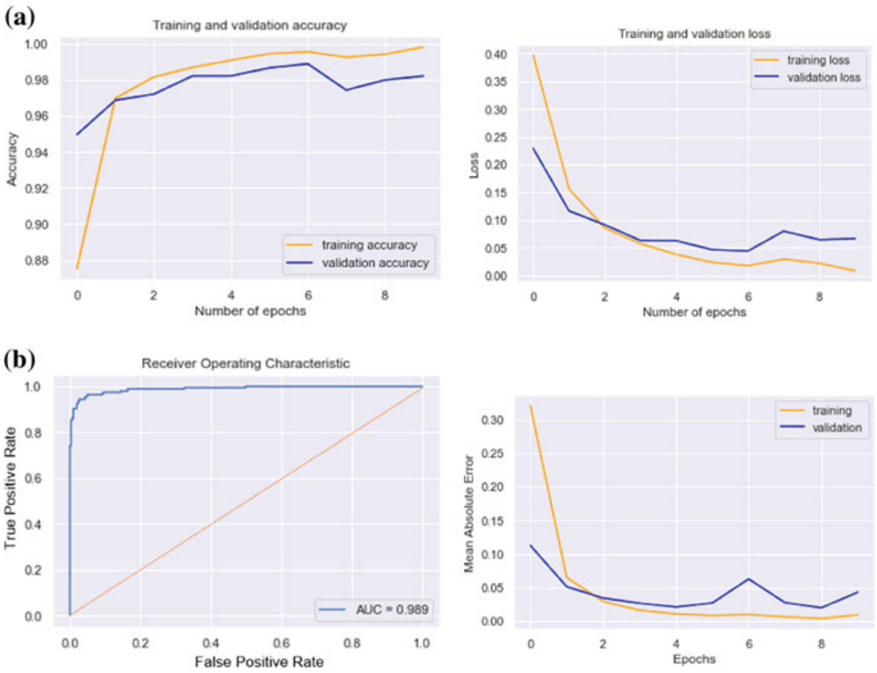


Fig. 4 **a** Spam prediction performance of the content model **b** ROC curve and MAE for content model

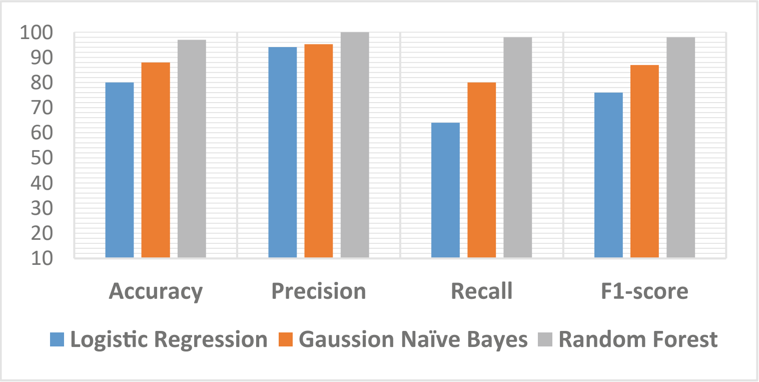


Fig. 5 Performance of bot detection

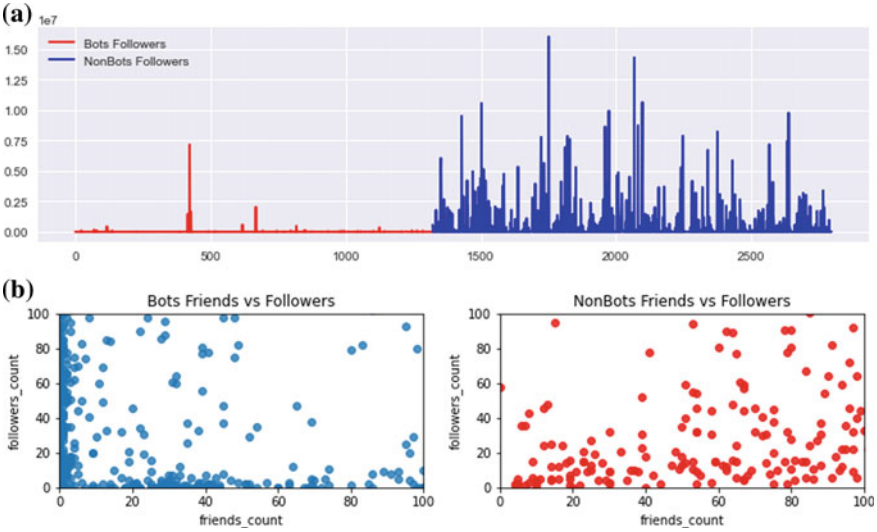


Fig. 6 **a** Bots followers vs non bots followers **b** friends vs following for bots and non-bots

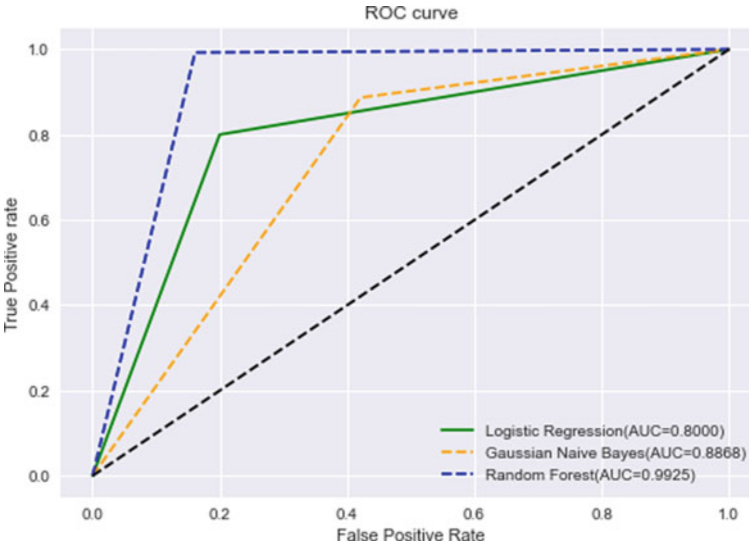


Fig. 7 ROC curve for bot classifier

Table 1 Comparative analysis of the proposed work

	Accuracy %	Precision %	Recall %	*F*-score %
Ensemble meta-classifier [24]	0.95	0.88	0.90	0.89
LSTM + attention [27]	88.00	0.88	0.88	0.88
LSTM + bot detection (proposed)	99.76	99.0	99.7	99.7

References

1. Rao, et al (2021) A reviews on social spam detection: Challenges, open issues, and future directions. Expert Syst Appl., 115742
2. Samper-Escalante et al (2021) Bot datasets on Twitter: Analysis and challenges. Appl Sci 11(9):4105
3. Ahmed E, et al (2020) Detecting spam in Twitter microblogging services: a novel machine learning approach based on domain popularity. Int J Adv Comput Sci Appl (IJACSA)
4. Ghanem et al (2020) Context-dependent model for spam detection on social networks. SN Appl Sci. 2(9):1–8
5. Deshmukh, Rushali (2021) Performance comparison for spam detection in social media using deep learning algorithms. Turk J Comput Math Educ (TURCOMAT) 12(1):193–201
6. Zhang, et al (2018) On scalable and robust truth discovery in big data social media sensing applications. IEEE Trans Big Data 5(2):195–208
7. Wu et al (2017) Twitter spam detection based on deep learning. In: 2017 Proceedings of the Australasian computer science week multiconference, 1–8
8. Jayashree P, Easwarakumar KS (2010) User behaviour trust model to defend denial of service attacks in distributed computational environments. Int J Comm Networks Distributed Syst 5(3):279–294
9. Badola et al (2021) Twitter spam detection using natural language processing by encoder decoder model. In: 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS). IEEE
10. Jardaneh et al (2019) Classifying Arabic tweets based on credibility using content and user features. In: 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT). IEEE
11. Singh et al (2020) Attention-based LSTM network for rumor veracity estimation of tweets. Inf Syst Front, 1–16
12. Viviani M, Pasi G (2017) Credibility in social media: opinions, news, and health information—a survey. Wiley Interdiscip Rev Data Min Knowl Discov 7(5):e1209
13. Alfred R, Teoh RW (2018) Improving topical social media sentiment analysis by correcting unknown words automatically. In: 2018 International Conference on Soft Computing in Data Science. Springer, Singapore
14. Kumar CSP, Dhinesh Babu LD (2019) Novel text preprocessing framework for sentiment analysis. In: 2019 Smart Intelligent Computing and Applications. Springer, Singapore
15. Stewart et al (2019) Word-level Lexical Normalisation using context-dependent embeddings. arXiv preprint arXiv:1911.06172
16. Masood et al (2019) Spammer detection and fake user identification on social networks. IEEE Access 7:68140–68152
17. Jenitha T, Jayashree P (2014) Distributed trust node selection for secure group communication in MANET. In: 2014 Fourth International Conference on Advances in Computing and Communications. IEEE
18. Loyola-Gonz et al (2019) Contrast pattern-based classification for bot detection on twitter. IEEE Access 7:45800–45817
19. Gadallah M, et al (2021) Credibility detection on Twitter news using machine learning approach. Int J Intell Syst Appl 13(3)
20. Sonawane D, Deepali, P, Gunjal, L (2020). New Approach for Detecting Spammers on Twitter using Machine Learning Framework. Int J Res Anal Rev (IJRAR). E-ISSN: 2348-1269
21. Zhang et al (2017) Semi-SGD: semi-supervised learning based spammer group detection in product reviews. In: 2017 Fifth International Conference on Advanced Cloud and Big Data (CBD), pp 368–373. IEEE
22. Etaiwi W, Arafat A (2017) The effects of features selection methods on spam review detection performance. In: 2017 International Conference on New Trends in Computing Sciences (ICTCS), pp 116–120

23. Jayashree P, et al (2022) Social network mining for predicting users' credibility with optimal feature selection. In: 2022 Intelligent Sustainable Systems. Springer, Singapore, pp 361–373
24. Madisetty et al (2018) A neural network-based ensemble approach for spam detection in Twitter. IEEE Trans Comput Soc Syst. 5(4):973–984
25. Dorri, et al (2018) SocialBotHunter: Botnet detection in Twitter-like social networking services using semi-supervised collective classification. In: 2018 16th International Conference on Dependable, Autonomic and Secure Computing, (DASC/PiCom/DataCom/CyberSciTech), pp 496–503. IEEE
26. Efthimion et al (2018) Supervised machine learning bot detection techniques to identify social twitter bots. SMU Data Sci Rev. 1(2):5
27. Ilias L et al (2021) Detecting malicious activity in Twitter using deep learning techniques. Appl Soft Comput 107:107360
28. AYDIN et al (2018) Detection of fake Twitter accounts with machine learning algorithms. In: 2018 International Conference on Artificial Intelligence and Data Processing (IDAP), pp 1–4

Face Recognition and Detection Algorithm—A Review

Puja S. Prasad, C. Esther Varma, and Vinit Kumar Gunjan

Abstract Biometric authentication is one of the more reliable authentication mechanism for identifying and verifying people. This is an automatic method to authenticate people using their biological characteristics like face, fingerprint, ear lobe, iris, etc. These biometric is divided into chemical, visual, vain and vascular, behavioral, and auditory depending upon nature of authentication. The main focus of this paper is to review about what are the different way of face recognition and detection as well as different classifier. Face detection is identifying whether face is present in the query image or not and is broader term then recognition. Face recognition means identifying the identity of a particular person.

Keywords Adaboost · Recognition · Detection · Gabor · LBP · PCA

1 Introduction

Face recognition is one of the most happening in biometrics research these days. Several places now generally have observation cameras for taking videos, and these cameras have their huge importance for security purpose [1]. It is broadly recognized that the face identification has played a crucial role in reconnaissance for number of day to day security purpose task [2]. The genuine benefits of facial recognition proof over other biometrics are their uniqueness and collectability. Facial feature is a unique article having changing features with age as well as its appearance that makes face identification a troublesome issue for authentication purpose. In this field, exactness and speed of recognizable proof is a principle issue [3]. The main goal of this paper is to evaluate and examine facial feature location and the outcome strategies, results total answer for picture-based face identification as well as recognition with higher precision, better recognizing rate in advancement of video recognition rate. Different algorithm are tested on different datasets rich in different race, feelings, color-based facial datasets [4].

P. S. Prasad (✉) · C. E. Varma · V. K. Gunjan
Geethanjali College of Engineering and Technology, Hyderabad, India
e-mail: puja.s.prasad@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_52

From last three decades, lots of work has been done for facial recognition as it does not require human participation. Facial features extraction is one of the most important steps in facial recognition. Biometric authentication and identification process as it only requires good datasets to create authentication model. Number of models have been developed for facial recognition system but it has number of challenges associated with it like different illumination conditions, different orientations and pose of images, other variational conditions, and very few number of datasets available for training. Face recognition uses different method for training in which one of them is One Shot Learning. One Shot Learning is trained by using very few set of datasets. Mostly, all organization in which facial recognition works apply this type of learning mechanism. An example of deep neural network architecture is FaceNet which uses batch input layer and deep CNN followed by L2 normalization which results in face embedding [5].

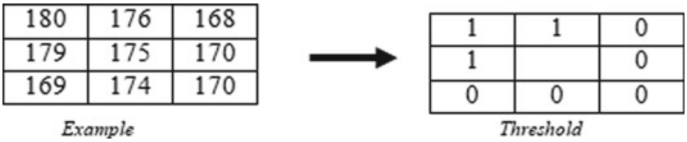
2 Face Detection

Combining classifier is also a very nice strategy to improve performance of classifier [6]. Adaboost combines number of weak classifiers to improve the performance of single weak classifier. Bagging stands for bootstrap aggregation. It averages over the predictions of all models. Most often bagging improves the performance of final classifier model. In bagging, different classifiers are trained on different datasets which randomly sampled from given data. Except for this random variations, the different classifiers effectively be the same. Boosting is major idea that come into pattern recognition over last fifteen years. In Adaboost, we assign weights to the point in the dataset which can be normalized so that they sum to one. Support vector machine is considered as one of the best classifier having great performance both in genomic to text data [7]. The popularity for SVM was gained during 1990s and can be applied to complex data types beyond feature vectors (e.g., graphs and relational data sequences) by designing kernel functions for such data. HAAR features are also used extracting features which uses HAAR filters. HAAR features are sensitive to directionality of patterns. Local binary pattern are used for detecting edges and one of the very popular technique and used in wide range of applications. LBP is efficient for texture classification. It codifies local pattern in which in which central pixels has some threshold and is compared with neighboring pixels. The Local binary pattern histograms have different subareas at that point, linked into a spatially improved part histogram and is characterized as:

$$H_{ij} =_{xy} (F_i(x, y) = i)I((x, y)$$

where $I = 0, \ldots, L{-}1; j = 0, \ldots, N{-}1$. The different separated element histogram portrays the nearby surface and datasets of facial features.

LBP computation. SVM classifier is been utilized with HOG highlights for face identification. Hoard enormously outflanks wavelets what's more, level of smoothing



180	176	168
179	175	170
169	174	170

1	1	0
1		0
0	0	0

Fig. 1 Pattern

prior to computing slopes harms, results underlines a significant part of the accessible data is from unexpected edges at fine scales that obscuring this for diminishing the affect-ability to spatial position is an error [8]. Angles ought to be determined at the best accessible scale in the current pyramid layer and solid nearby differentiation standardization is fundamental for acceptable outcomes [6, 9] (Figs. 1 and 2).

Though SVM are planned to settle an old style two class issue which returns a parallel worth, the class of the object. To prepare our SVM calculation, we plan the

Fig. 2 Face detection. *Source* Internet

issue in a distinction space that expressly catches the uniqueness between two facial pictures [10].

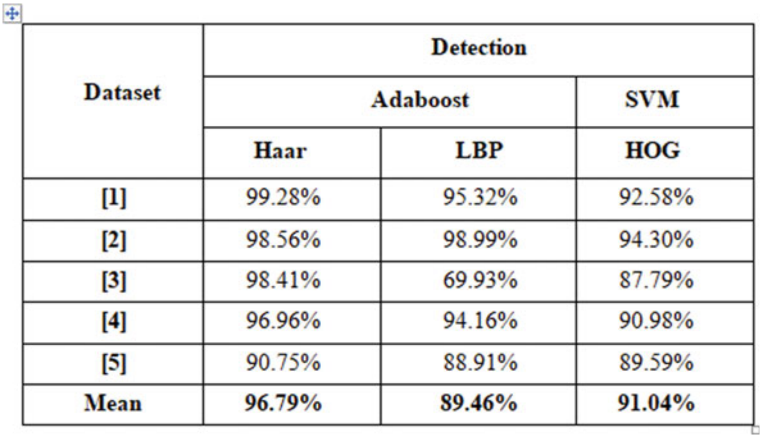
3 Face Recognition

Eigenfaces considered as 2D face acknowledgment issue, and countenances will be generally upstanding and front facing [11]. That is the reason three dimensional data about the face isn't needed that lessens intricacy by a huge piece. It converts the face pictures into a bunch of premise capacities which basically are the head parts of the face pictures looks for headings in which it is more productive to address the information [12]. This is mostly valuable for decline the computational exertion (Fig. 3).

Direct discriminant examination is principally utilized here to decrease the number of elements to a more sensible number previously acknowledgment since face is addressed by an enormous number of pixel esteems. Every one of the new aspects is a straight mix of pixel esteems, which structure a format. The direct blends acquired utilizing Fisher's straight discriminant are called Fisherfaces. LBP is an request set of double examinations of pixel forces between the middle pixel and its eight encompassing pixels [13].

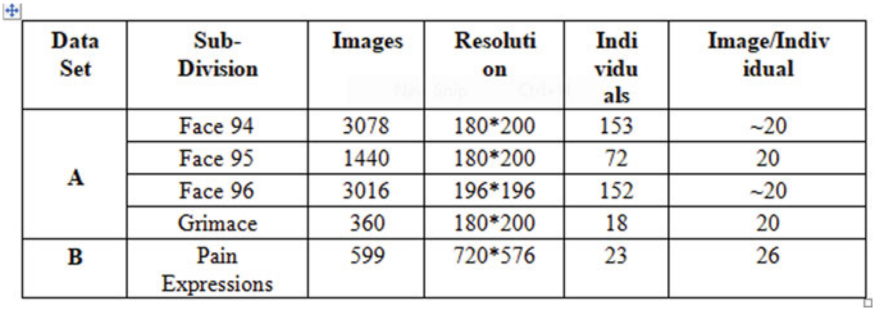
$$\mathrm{LBP}(x_a, y_a) = \sum_{n} oS(i_m - i_a)2^n$$

Gabor channels can take advantage of striking visual properties such as spatial restriction, direction selectivity, and spatial recurrence qualities [14]. Thinking about



Dataset	Detection		
	Adaboost		SVM
	Haar	LBP	HOG
[1]	99.28%	95.32%	92.58%
[2]	98.56%	98.99%	94.30%
[3]	98.41%	69.93%	87.79%
[4]	96.96%	94.16%	90.98%
[5]	90.75%	88.91%	89.59%
Mean	**96.79%**	**89.46%**	**91.04%**

Fig. 3 Face detection results summary



Data Set	Sub-Division	Images	Resolution	Individuals	Image/Individual
A	Face 94	3078	180*200	153	~20
	Face 95	1440	180*200	72	20
	Face 96	3016	196*196	152	~20
	Grimace	360	180*200	18	20
B	Pain Expressions	599	720*576	23	26

Fig. 4 Face Recognition results summary

these overwhelming limits and its extraordinary accomplishment in face acknowledgment, Gabor highlights are obtuse toward changes as enlightenment, posture, and articulations in spite of the fact that Gabor change isn't extraordinarily intended for face acknowledgment [15]. Its change equation is predefined rather than learned from the face preparing information. Also PCA and LDA classifier think about worldwide highlights while LBP furthermore, Gabor classifier think about neighborhood highlights, in light of current realities exploratory outcomes are expressed below.

4 Conclusion

This paper discussed about difference between face recognition and detection. Face detection is done by using Adaboost. Face detection is detecting one or many faces in steal images or sequence of video images. Eigenface is image-based detection which involves the extraction of eigenfaces by using Principal Component analysis which reduces the dimensionality of input spaces. One of the problem with eigenfaces is that it does not minimize intra class variance. Fisher's Linear Discriminant is classifier is optimal classifier compared to PCA which reduces intraclass variance. Machine learning algorithm also provides method to train classifier for detection and recognition of faces using unique measurement of faces and match that data with the known faces in a database. Kernel method and SVM are some of the machine learning algorithm (Fig. 4).

References

1. Ahonen T, Hadid A, Pietik¨ainen M (2004) Face recognition with local binary patterns. In: European conference on computer vision, pp 469–481. Springer
2. Belhumeur PN, Hespanha JP, Kriegman DJ (1997) Eigenfaces vs. fisherfaces: recognition using class specific linear projection. IEEE Trans Pattern Anal Mac Intell 19(7):711–720

3. Brunelli R, Poggio T (1993) Face recognition: features versus templates. IEEE Trans Pattern Anal Mach Intell 15(10):1042–1052
4. He X, Yan S, Yuxiao H, Niyogi P, Zhang H-J (2005) Face recognition using laplacianfaces. IEEE Trans Pattern Anal Mach Intell 27(3):328–340
5. Jafri R, Arabnia HR (2009) A survey of face recognition techniques. J Info Proces Syst 5(2):41–68
6. Schapire RE (2013) Explaining adaboost. In: Empirical inference. Springer, pp 37–52
7. Lin S-H (2000) An introduction to face recognition technology. Informing Sci Int J Emerg Transdiscipl 3:1–7
8. Faruqe MO, Al Mehedi Hasan M (2009) Face recognition using pca and svm. In: 2009 3rd International Conference on Anti-counterfeiting, Security, and Identification in Communication, pp 97–101. IEEE
9. Lihong Z, Ying S, Yushi Z, Cheng Z, Yi Z (2009) Face recognition based on multi-class svm. In: 2009 Chinese Control and Decision Conference. IEEE, pp 5871–5873
10. Kim S-K, Park YJ, Toh K-A, Lee S (2010) Svm-based feature extraction for face recognition. Pattern Recog 43(8):2871–2881
11. Turk M, Pentland A (1991) Eigenfaces for recognition. J Cogn Neurosci 3(1):71–86
12. Schneier B (1999) The uses and abuses of biometrics. Commun ACM 42(8):136–136
13. Phillips P (1998) Support vector machines applied to face recognition. Adv Neural Inf Process Syst 11:803–809
14. Uddin MN, Sharmin S, Ahmed AH, Hasan E (2011) A survey of biometrics security system. IJCSNS 11(10):16
15. Turk MA, Pentland AP (1991) Face recognition using eigenfaces. In: Proceedings. 1991 IEEE computer society conference on computer vision and pattern recognition, pp 586–587. IEEE Computer Society

A Survey of Ciphertext Processing Techniques

Sourabh Bhaskar, Keyur Parmar, and Devesh C. Jinwala

Abstract The demand for data privacy is rapidly increasing with the growing volume of personal and sensitive information shared to compute mathematical functions. Some of the traditional data encryption techniques provide data privacy but fail in hassle-free sharing. Therefore, the third-party users can not perform mathematical computations over the data to generate meaningful outcomes. In this situation, ciphertext processing techniques are used. Ciphertext processing techniques provide computations over encrypted data. The survey discusses the ciphertext processing techniques that enable mathematical computations over the encrypted data to third-party users while maintaining data privacy. The ciphertext processing techniques also enable privacy-preserving data analysis using machine learning or statistical tools. In addition, we discuss the applications of ciphertext processing techniques. We present the research gaps in the domain of ciphertext processing and provide future research directions.

Keywords Privacy · Homomorphic encryption · Secure multi-party computation · Secure searchable encryption · Cloud

1 Introduction

The privacy concerns to protect the data collected from various application areas such as healthcare, finance, businesses, are increasing day by day. Therefore, encryp-

S. Bhaskar (✉) · K. Parmar · D. C. Jinwala
S. V. National Institute of Technology, Surat, India
e-mail: sourabhhb440@gmail.com

K. Parmar
e-mail: keyur@coed.svnit.ac.in

D. C. Jinwala
e-mail: dcj@svnit.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_53

tion is a superior technique to preserve the privacy of crucial data [1]. The encryption could be helpful in preserving privacy from unauthorized extraction of sensitive and personal information from the cloud. The encrypted data are easily stored or retrieved from the cloud to the authorized users. However, the limitation with encrypted data is not able to perform computations [2]. Therefore, the requirement for such privacy-preserving computable techniques occurs. These techniques can perform the insightful computation over encrypted data without decrypting it, and generated results should match with the computation performed on plaintext data called "privacy homomorphism" came in 1978 [1].

The ciphertext processing techniques are those methods that enable the third-party to perform the functional computation over encrypted data. The technique works without knowing anything about the data and obtains encrypted results without compromising the privacy of an individual [3]. After decrypting, the result is the same as while doing computation on normal plaintext. Let's assume msg_1 and msg_2 are two messages.

$$c_1 = \mathrm{Encr}(\mathrm{pk}, \mathrm{msg}_1)$$

$$c_2 = \mathrm{Encr}(\mathrm{pk}, \mathrm{msg}_2)$$

$$\mathrm{msg}_1 + \mathrm{msg}_2 = \mathrm{Decr}(\mathrm{sk}, c_1 \oplus c_2)$$

In this, c_1 and c_2 are ciphertexts of msg_1 and msg_2 respectively; Encr and Decr are encryption and decryption algorithms; The pk and sk denotes public key and private key, respectively. We have several ciphertext processing techniques available to process the ciphertext to compute meaningful outcomes such as Homomorphic Encryption (HE), Secure Multi-Party Computation (SMPC), and Searchable Symmetric Encryption (SSE).

As a sample example of medicine and bioinformatics, ethical concerns and privacy regulations are available to prevent the sharing of concern data. These regulations make it challenging to utilize the computational facilities without compromising the privacy of the stored data. Ciphertext processing techniques facilitate privacy-enabled sharing of encrypted medical data and allow machine learning and statistical techniques over encrypted data [4]. Ciphertext processing techniques allow multiple parties to compute the function without revealing their data inputs jointly [5]. Ciphertext processing techniques resolve many problems such as Yao's Millionaires' problem [5], Mental poker [6], Electronic voting, Auction.

In today's era, data is essential. When we were walking down the street and needed coffee, we searched on mobile GPS nearby coffee shops. In such query, we reveal the entities, such as our location (where we are), time, and date of questioning, and we want a coffee. Thus, such data and questions are required to be encrypted. The entity performing the computation should not know anything about the data but performs the match and returns the match [7].

The paper aims to discuss the concept and significance of ciphertext processing techniques. We focus on the role of different ciphertext processing techniques such as HE, SMPC, and SSE. Ciphertext processing techniques can adapt in different domains to enable privacy preservation based on data sharing and computing capabilities over encrypted data. In addition, we discuss the benefits and the challenges of the Fully Homomorphic Encryption (FHE) scheme and provide future research directions.

2 Homomorphic Encryption

We discuss the essentials of HE. Then, we describe different HE schemes. In addition, we provide a concise description of each scheme.

The term homomorphic was first coined by Rivest et al. [1] in 1978 with the conceptualization of "privacy homomorphism". HE focuses on privacy-enabled computation over encrypted data without being decrypted. The properties of homomorphically additive and multiplicative schemes are given below:

$$[p] \oplus [q] = [p + q]$$

$$[p] \otimes [q] = [p * q]$$

The above mentioned equations are the example of homomorphic addition and homomorphic multiplication, respectively. While $[p]$ represents the encrypted form of plaintext message p. The symbols $\oplus$ and $\otimes$ denote the notation of homomorphic addition and multiplication operations, respectively, in the ciphertext. A HE scheme can be defined into three categories based on the number of operations usable by the scheme, and Fig. 1 shows the HE scheme's timeline.

- Homomorphic schemes permit to execute only individual operation either addition or multiplication, on the encrypted data along with arbitrary times use, called partially homomorphic encryption scheme (PHE).
- Homomorphic schemes permit to execute set of operations (addition and/or multiplication), on the encrypted data along with limited times use, called Somewhat homomorphic encryption scheme (SWHE).

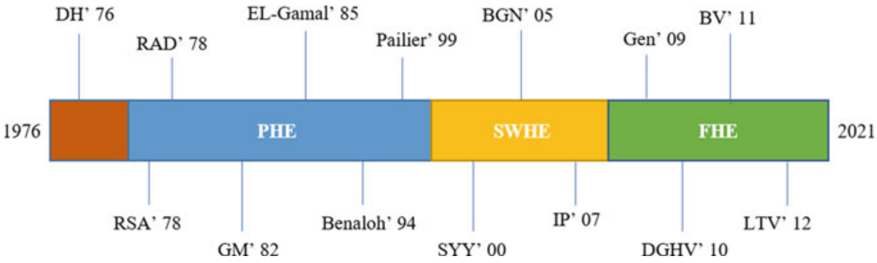


Fig. 1 Timeline of homomorphic encryption schemes [3]

Table 1 Notable partially homomorphic encryption schemes

Authors	Cryptosystems	Homomorphic operation
Rivest et al. [9]	RSA cryptosystem	Multiplication
Goldwasser and Micali [10]	GM cryptosystem	Addition
ElGamal [11]	ElGamal cryptosystem	Multiplication
Paillier [12]	Paillier cryptosystem	Addition

- Homomorphic schemes permit to execute both operations (addition and multiplication), on the encrypted data along with arbitrary times use, called Fully homomorphic encryption scheme (FHE).

2.1 Partially Homomorphic Encryption

The PHE scheme permits individual operations (addition or multiplication) with an arbitrary number of times over encrypted data. PHE-based schemes are efficient to implement but support a controlled number of mathematical operations over encrypted data [8]. We discuss some of the useful PHE schemes as shown in Table 1.

RSA Cryptosystem Subsequently, the public key encryption (PKE) was discovered by Diffie and Hellman [13], and Rivest et al. [9] proposed the RSA scheme. The homomorphic property of RSA defines that $\mathrm{Encr}(\mathrm{msg}_1 * \mathrm{msg}_2)$ is straightforwardly assessed through utilizing $\mathrm{Encr}(\mathrm{msg}_1)$ and $\mathrm{Encr}(\mathrm{msg}_2)$ without decrypting them. The strength of the RSA algorithm is defined on the hardness of factorization of chosen two large prime numbers [14]. RSA defined three algorithms such as key generation, encryption, and decryption.

GM Cryptosystem Goldwasser and Micali [9] proposed the principal probabilistic PKE scheme. The GM cryptosystem relies upon the hardness of the quadratic residuosity problem. The GM cryptosystem is homomorphic on addition operation.

ElGamal Cryptosystem ElGamal [11] proposed a PKE scheme. The PKE scheme was the enhanced version of the initial Diffie-Hellman key exchange. The ElGamal cryptosystem does not support additive homomorphism and works with multiplication operations. The ElGamal cryptosystem depends on the hardness of the discrete logarithm.

Paillier Cryptosystem Paillier [12] proposed a probabilistic encryption scheme based on the composite residuosity problem. Paillier's cryptosystem is homomorphic over addition operation.

2.2 Somewhat Homomorphic Encryption

SWHE scheme permits to perform some set of operations a limited number of times. We discuss some of the useful SWHE schemes as shown in Table 2.

Table 2 Notable somewhat homomorphic encryption schemes

Authors	SWHE schemes
Sander et al. [15]	SYY
Boneh et al. [16]	BGN
Ishai and Paskin [17]	IP

Table 3 Notable fully homomorphic encryption schemes

Authors	FHE schemes
Gentry [18]	Ideal lattice based FHE scheme
Dijk et al. [20]	Integer based FHE scheme
Brakerski and Vaikuntanathan [21]	LWE/R-LWE based FHE scheme
López-Alt et al. [22]	NTRU based FHE scheme

SYY Sander et al. [15] proposed the SWHE scheme over a semigroup, NC_1. The SYY scheme supports multiple AND and one OR gate. The constant ANDing with each OR gate evaluation has increased the size of the ciphertext.

BGN Boneh et al. [16] define the evaluation of 2-DNF formulas on the ciphertext. The BGN scheme accepted the arbitrary number of addition and one multiplication. BGN scheme figure out quadratic multi-variate polynomials on ciphertexts gave the subsequent worth falls inside a little set. The security of the BGN scheme depends on the subgroup decision issue.

IP Ishai and Paskin [16] proposed a PKE scheme on encrypted data through a branching program. The resultant ciphertext size of the IP scheme does not rely on the function size. Then we evaluate decision trees and finite automata efficiently. The evaluation circuit used by the IP scheme is a branching program.

2.3 Fully Homomorphic Encryption

The first FHE scheme was proposed by Gentry [18, 19] in 2009 in his thesis work. Gentry's presented scheme is not only a FHE but also the generic structure to build a FHE scheme. Thus, lots of research attempts were applied to design a more privacy enable and feasible FHE scheme afterward gentry's scheme [2, 3]. The limitation of the FHE schemes is challenging to practically implement due to their high computation overhead (Table 3).

Fully Homomorphic Encryption based on Ideal Lattices Gentry [18] proposed a FHE scheme based on an ideal lattice in his Ph.D. thesis in 2009. The SWHE scheme performs only a fixed number of operations over encrypted data. When the limit surpasses, the decrypting algorithm fails to recover the processed information from ciphertext precisely. The volume of noise should not be more than the threshold so that noisy ciphertext can transform easily into a normal ciphertext.

Gentry's FHE scheme based on an ideal lattice result has been divided into three steps: an "initial construction" using ideal lattice, a general "bootstrapping" result, "squash the decryption circuit" to permit bootstrapping. A homomorphic PKE scheme ε has four algorithms as follows: KeyGen$_{\varepsilon}$, Encrypt$_{\varepsilon}$, Decrypt$_{\varepsilon}$, and Evaluate$_{\varepsilon}$.

Fully Homomorphic Encryption based on Integer Dijk et al. proposed a FHE over integer using SHWE scheme, and the Approximate Greatest Common Divisors (AGCD) problems describe the hardness of the scheme [20]. FHE based on the integer scheme was the symmetric version of homomorphic encryption and was conceptually easy than others.

Fully Homomorphic Encryption based on Learning With Errors Brakerski and Vaikuntanathan [21] introduced another SWHE scheme dependent on RLWE to take benefit of the well-defined feature of RLWE. Specifically, the hardness assumption of an FHE used both LWE and RLWE. RLWE shows better efficiency.

NTRU—like Fully Homomorphic Encryption Lopez-Alt et al. [22] presented SMPC through a robust computation enabled but not secure third-party cloud. NTRU allows the arbitrary sets of parties on the fly to perform computation on the related data. Authors construct of NTRU based multi-key FHE scheme by employing the method of modulo reduction and re-linearization.

2.4 Applications

These days, homomorphic encryption is mainly introduced to provide privacy preserved on cloud-based applications or platforms [23]. Therefore, we are going to list some of the well-known platforms where authors have implemented the HE schemes given below:

- Ren et al. [24] used homomorphic encryption for improving the security and privacy of cloud-based mobile IoT systems.
- To reduce the energy consumption, data integration is an important method in wireless sensor networks; thus, Othman et al. [25] try to perform data integration while maintaining data privacy and integrity.
- Hirt and Sako [26] proposed the first homomorphic encryption-based receipt free practical efficient voting system.
- A privacy-preserving integration introduced in the Internet of things using homomorphic encryption. The privacy-preserving integration is applicable on hop-to-hop and count-to-count layering on cloud environments [23, 27].
- Machine learning and statistical tools are powerful tools for analyzing the facts and results. FHE schemes are used in medicine and bioinformatics to perform arbitrary

computations using machine learning in different forms of healthcare data such as heart rate, respiratory rate, and temperature [4].

- Blake et al. [28] discuss next-generation data-sharing using privacy-enhanced homomorphic encryption schemes. Homomorphic encryption is the most suitable technique while preserving their privacy to analyze the risk and statistical data to visualize the results in financial services.
- Gupta and Arora [7] use homomorphic encryption along with GPS to provide location-based privacy to the users.

3 Secure Multi-party Computation

SMPC is one of the methods of privacy-preserving computation where multiple parties over their inputs together compute a function without sharing their inputs to others. SMPC protects participants' input data privacy. There are many kinds of computation which are listed in Table 4.

SPC protocol was introduced for a specific purpose in the late 1970s [6]. Later, to perform secure computation, two or multiple-party computation protocol formally introduced in 1982 by Yao's seminal [5] and in generality 1986 [29]. Presently, the developing prevalence of the recently arising advancements like IoT, distributed or mobile computing has brought about a re-birth of SMPC's inescapability.

In SMPC, multiple parties $m_j = (j = 1, 2, 3..., n)$ with their private inputs p_j, in the distributed computing environment want to interactively and jointly compute over an objective function f(p_1, p_2, p_3, ..., p_n) = (q_1, q_2, q_3,..., q_n) depends on their given private inputs. After the completion of computation, all party takes their own output q_j without knowing about others information.

SMPC permits a bunch of players to together process a random function of players' inputs without revealing the player's inputs to one another. For Example, the players can register the average of their salaries so that no player learns the pay of some other

Table 4 Comparision of different SMPC computation protocols

Secure computations	Protocols	Applications
Secure two party computation (S2PC)	Yao's protocol [5], Garbled circuit protocol [29], 1-out-of-2 oblivious transfer protocol [30]	E-commerce and data mining
Secure multi-party computation (SMPC)	Shamir secret sharing [30], Additive secret sharing, and SPDZ [31]	Genetic tests and signature sharing
Special purpose computation (SPC)	Mental poker [6]	Voting, auction, and payments

player. Or on the other hand, as a more applied model, many citizens can figure the sum of their votes without uncovering a specific vote.

Bentov and Kumaresan [32] presented a model for secure computation in which adversary aborts or restricts to receive output and results in opting predefined monetary for fairness. The model [32] has been discussed for the bitcoin network and the fair lottery.

3.1 Applications

Bogetoft et al. [33] discuss the execution of a safe exchanging arrangement of specific products among merchants and purchasers called double auction. The framework was completed using SMPC. SMPC guarantees each bid submitted to the auction is kept encoded in the given time, and no parties have permission to access the bids at another time. Thus, the framework could productively process the cost for which agreement to be done.

SMPC protocols are used for preserving privacy based on various genetic tests [34]. Some different utilizations of SMPC such as appropriated private bidding and auctions, distributed voting, sharing of signatures, or unscrambling private data. Various frameworks have carried out different types of SMPC with secret sharing plans. The most well-known framwork is SPDZ. Damgard et al. [31] carries out SMPC with secret additive offers and is secure against agile enemies. In addition, SMPC protocols are adopted by the secure genomic sequence comparison for genomic sequence computation, Private set operations, cloud-oriented applications, SMPC on big data, etc.

4 Searchable Symmetric Encryption

SSE empowers customers to send their encoded data for storage on the third-party cloud/server in a private manner. In contrast, keeping up the ability of keyword searches without uncovering data about the substance of records over encoded data [35, 36].

Here users locally encrypt their data then outsource on server/cloud. The server does not know about the content of data due to encryption (Fig. 2). Whenever users want to access their data, users directly search on encrypted data by keyword [37]. A limitation of SSE is revocation which is not efficiently manageable in the proposed scheme [38]. Later on, different improved SSE were proposed in terms of better efficiency for search computation on large-scale datasets [39, 40].

Cash et al. [41] plan and carry out unique symmetric accessible encryption conspire that effectively and secretly search workers held encrypted data sets with a huge number of records. Our fundamental theoretical advancement maintains single-

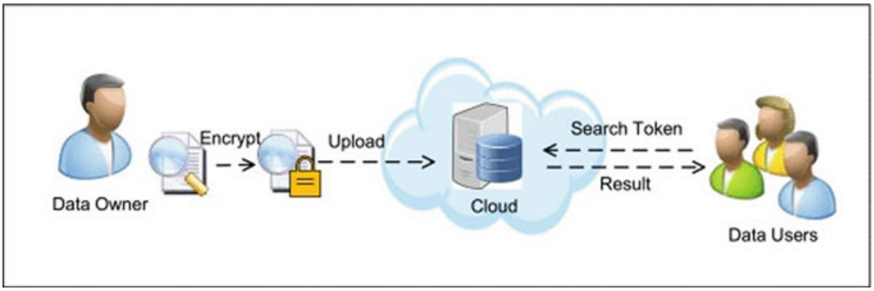


Fig. 2 Searchable encryption model for application systems [43]

keyword searches and offers asymptotically ideal server token size, completely equal searching, and negligible leakage.

For electronic health records (EHRs), Chen et al. [42] presented an accessible encryption system based on blockchain technology. The EHRs are created through complex rationale articulations and set aside in the blockchain. The objective is that a client utilizes the explanations to glance through the document. As the record is moved to the blockchain to work with expansion, the information proprietor has full order over who can see their EHRs data. The use of blockchain development ensures the honesty, hostile to altering, and recognizability of EHRs' records. In the end, the suggested method's demonstration is evaluated from two angles, specifically in relation to the overhead for deleting the record IDs from electronic health records and the overhead for carrying out transactions on Ethereum smart contracts.

4.1 Applications

Zhang et al. [43] define that searchable encryption with the accompanying two capacities for ensuring information protection and access security: (1) Need to pass on the encoded information to approved clients and empower querying over scrambled information, and (2) they likewise need to keep the query keywords and related trail tasks private.

Li et al. [44] proposed two effective and secure dynamic searchable encryption techniques over sensitive healthcare cloud data. Where the patient data can be remotely outsourced on the cloud in the encrypted form and accessed by only the hospital's authorized staff. Li et al. [44] have compared their schemes with traditional schemes and found superior in terms of storage, search, and updating efficiency of the algorithm.

5 Problem Statement and Research Gaps

When some cryptographic algorithms are applied encryption algorithms to plaintext data, the size of ciphertext increases more as compared to the original plaintext. Ciphertext processing techniques may comprise with few amount of noise in the ciphertext. The ciphertext can be accurately decrypted if the noise is up to the threshold value.

FHE schemes broadly increase the scope of computation over the cloud. The cloud analyses homomorphically encrypted data on behalf of the user and returns the encrypted result. However, the FHE schemes are practically expansive in terms of performance for real-world applications. A further suggestion is to implement the FHE schemes with various open-source libraries and compare them with different cryptographic techniques, which may have homomorphic properties. Improvements in the performance and usability of FHE schemes may introduce new applications to be implemented.

SMPC addresses a wide range of research topics, from theoretical to practical perspectives. Secure genomic sequence comparison majorly adopted the S2PC protocol for secure genomic computation. SMPC can be more efficiently explored in privacy-preserving data queries and data mining.

In addition, the SSE needs to be devised a new security model for emerging attacks.

6 Conclusions and Future Research Directions

We discussed different ciphertext processing techniques such as homomorphic encryption schemes, secure multi-party schemes, searchable symmetric encryption schemes, and timeline-wise development reviews of the homomorphic encryption schemes. The ability to perform the mathematical and functional computations on encrypted data without knowing about the data contents. Ciphertext processing techniques lead to a rapid implementation from various backgrounds to utilize ciphertext processing techniques for the research. Ciphertext processing techniques enable the use of machine learning and statistical tools to analyze prediction-based outcomes. Ciphertext processing techniques effectively protect data privacy on multiple platforms. Ciphertext processing techniques solve open research problems related to privacy in the future when introduced on several platforms. In this paper, we discussed different application developments of all three techniques and the current possibilities. We also provided the challenging research issues for FHE related to its efficient implementation.

References

1. Rivest RL, Adleman L, Dertouzos ML et al (1978) On data banks and privacy homomorphisms. Found Secure Comput 4(11):169–179. https://luca-giuzzi.unibs.it/corsi/Support/papers-cryptography/RAD78.pdf. Accessed 10 Nov 2021
2. Fontaine C, Galand F (2007) A survey of homomorphic encryption for nonspecialists. EURASIP J Inf Secur 1:1–10. https://doi.org/10.5555/2907333.2907524
3. Acar A, Aksu H, Uluagac AS, Conti M (2018) A survey on homomorphic encryption schemes: theory and implementation. ACM Comput Surv 51(4):1–35. https://doi.org/10.1145/3214303
4. Wood A, Najarian K, Kahrobaei D (2021) Homomorphic encryption for machine learning in medicine and bioinformatics. ACM Comput Surveys (CSUR) 53(4):1–35. https://doi.org/10.1145/3394658
5. Yao AC (1982) Protocols for secure computations. In: 23rd annual symposium on foundations of computer science. SFCS, IEEE, Chicago, USA, pp 160–164, Nov 1982. https://doi.org/10.1109/SFCS.1982.38
6. Shamir A, Rivest RL, Adleman LM (1981) Mental poker. In: The mathematical gardner. Springer, pp 37–43. https://doi.org/10.1007/978-1-4684-6686-7_5
7. Gupta S, Arora G (2019) Use of homomorphic encryption with GPS in location privacy. In: Proceedings of the 4th international conference on information systems and computer networks (ISCON). IEEE, Mathura, India, pp 42–45, March 2019. https://doi.org/10.1109/ISCON47742.2019.9036149
8. Shoukry Y, Gatsis K, Alanwar A, Pappas GJ, Seshia SA, Srivastava M, Tabuada P (2016) Privacy-aware quadratic optimization using partially homomorphic encryption. In: Proceedings of the 55th conference on decision and control. IEEE, Las Vegas, USA, pp 5053–5058, Dec 2016. https://doi.org/10.1109/CDC.2016.7799042
9. Rivest RL, Shamir A, Adleman L (1978) A method for obtaining digital signatures and public-key cryptosystems. Commun ACM 21(2):120–126. https://doi.org/10.1145/359340.359342
10. Goldwasser S, Micali S (1982) Probabilistic encryption and how to play mental poker keeping secret all partial information. In: Proceedings of the 14th annual ACM symposium on theory of computing. STOC'82, Association for Computing Machinery, pp 365–377, May 1982. https://doi.org/10.1145/800070.802212
11. ElGamal T (1985) A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans Inf Theory 31(4):469–472. https://doi.org/10.1007/3-540-39568-7_2
12. Paillier P (1999) Public-key cryptosystems based on composite degree residuosity classes. In: Proceedings of the international conference on the theory and applications of cryptographic techniques. EUROCRYPT 1999, Springer Berlin Heidelberg, Prague, Czech Republic, pp 223–238, May 1999. https://doi.org/10.1007/3-540-48910-X_160
13. Diffie W, Hellman M (1976) New directions in cryptography. IEEE Trans Inf Theory 22(6):644–654. https://doi.org/10.1109/TIT.1976.1055638
14. Montgomery PL (1994) A survey of modern integer factorization algorithms. CWI Q 7(4):337–365. https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.32.2831&rep=rep1&type=pdf. Accessed 10 Nov 2021
15. Sander T, Young A, Yung M (1999) Non-interactive cryptocomputing for NC/SUP 1. In: Proceedings of the 40th annual symposium on foundations of computer science. IEEE, New York, USA, pp 554–566, Oct 1999. https://doi.org/10.1109/SFFCS.1999.814630
16. Boneh D, Goh EJ, Nissim K (2005) Evaluating 2-DNF formulas on ciphertexts. In: Theory of cryptography conference, vol 3378. Springer, Cambridge, USA, pp 325–341, Feb 2005. https://doi.org/10.1007/978-3-540-30576-7_18
17. Ishai Y, Paskin A (2007) Evaluating branching programs on encrypted data. In: Theory of cryptography conference. TCC 2007, Springer, Amsterdam, The Netherlands, pp 575–594, Feb 2007. https://doi.org/10.1007/978-3-540-70936-7_31
18. Gentry C (2009) Fully homomorphic encryption using ideal lattices. In: Proceedings of the 41st annual ACM symposium on Theory of computing. STOC '09, Association for Computing

Machinery, Bethesda MD, USA, pp 169–178, May 2009. https://doi.org/10.1145/1536414.1536440

19. Gentry C (2021) A fully homomorphic encryption scheme, vol 20. Stanford university, Stanford, pp 1–209, Sept 2009. https://crypto.stanford.edu/craig/craig-thesis.pdf. Accessed 10 Nov 2021
20. Van Dijk M, Gentry C, Halevi S, Vaikuntanathan V (2010) Fully homomorphic encryption over the integers. In: Proceedings of the annual international conference on the theory and applications of cryptographic techniques. Eurocrypt 2010, Springer, Monaco and Nice, France, pp 24–43, June 2010. https://doi.org/10.1007/978-3-642-13190-5_2
21. Brakerski Z, Vaikuntanathan V (2011) Fully homomorphic encryption from ring-LWE and security for key dependent messages. In: Advances in cryptology—CRYPTO 2011. Springer, Santa Barbara, USA, pp 505–524, Aug 2011. https://doi.org/10.1007/978-3-642-22792-9_29
22. López-Alt A, Tromer E, Vaikuntanathan V (2012) On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption. In: Proceedings of the 44th annual ACM symposium on theory of computing. STOC'12, Association for Computing Machinery, New York, USA, pp 1219–1234, May 2012. https://doi.org/10.1145/2213977.2214086
23. Min Zhao E, Geng Y (2019) Homomorphic encryption technology for cloud computing. Proceedings of the 9th international conference of information and communication technology (ICICT), vol 154. Elsevier, Nanning, Guangxi, China, pp 73–83, Jan 2019. https://doi.org/10.1016/j.procs.2019.06.012
24. Ren W, Tong X, Du J, Wang N, Li SC, Min G, Zhao Z, Bashir AK (2021) Privacy-preserving using homomorphic encryption in Mobile IoT systems. Secure Artif Intell Mobile Edge Comput 165:105–111. https://doi.org/10.1016/j.comcom.2020.10.022
25. Othman SB, Bahattab AA, Trad A, Youssef H (2015) Confidentiality and integrity for data aggregation in WSN using homomorphic encryption. Wireless Personal Commun 80(2):867–889. https://doi.org/10.1007/s11277-014-2061-z
26. Hirt M, Sako K (2000) Efficient receipt-free voting based on homomorphic encryption. In: Proceedings of the international conference on the theory and applications of cryptographic techniques. EUROCRYPT 2000, Springer, Lecture notes in computer science, Bruges, Belgium, pp 539–556, May 2000. https://doi.org/10.1007/3-540-45539-6_38
27. Zouari J, Hamdi M, Kim TH (2017) A privacy-preserving homomorphic encryption scheme for the internet of things. In: 13th international wireless communications and mobile computing conference (IWCMC). IEEE, Valencia, Spain, pp 1939–1944, June 2017. https://doi.org/10.1109/IWCMC.2017.7986580
28. The next generation of data-sharing in financial services using privacy enhancing techniques to unlock new value. World Economic Forum, pp 1–34 (2019). https://www2.deloitte.com/content/dam/Deloitte/lu/Documents/financial-services/lu-next-generation-data-sharinging-financial-services.pdf. Accessed 18 Oct 2021
29. Yao ACC (1986) How to generate and exchange secrets. In: Proceedings of the 27th annual symposium on foundations of computer science. SFCS 1986, IEEE, Toronto, Canada, pp 162–167, Oct 1986. https://doi.org/10.1109/SFCS.1986.25
30. Rabin MO (2005) How to exchange secrets with oblivious transfer. IACR Cryptol ePrint Arch 2005(187)
31. Damgård I, Pastro V, Smart N, Zakarias S (2012) Multiparty computation from somewhat homomorphic encryption. In: Annual cryptology conference. Springer, Santa Barbara, USA, pp 643–662, Aug 2012. https://doi.org/10.1007/978-3-642-32009-5_38
32. Bentov I, Kumaresan R (2014) How to use Bitcoin to design fair protocols. In: Annual cryptology conference. Springer, Santa Barbara, USA, pp 421–439, Aug 2014. https://doi.org/10.1007/978-3-662-44381-1_24
33. Bogetoft P, Christensen DL, Damgård I, Geisler M, Jakobsen T, Krøigaard M, Nielsen JD, Nielsen JB, Nielsen K, Pagter J (2009) Secure multiparty computation goes live. In: International conference on financial cryptography and data security. Springer, Accra Beach, Barbados, pp 325–343, Feb 2009. https://doi.org/10.1007/978-3-642-03549-4_20

34. Dugan T, Zou X (2016) A survey of secure multiparty computation protocols for privacy preserving genetic tests. In: Proceedings of the first international conference on connected health: applications, systems and engineering technologies (CHASE). IEEE, Washington, DC, USA, pp 173–182, June 2016. https://doi.org/10.1109/CHASE.2016.71
35. Curtmola R, Garay J, Kamara S, Ostrovsky R (2006) Searchable symmetric encryption: improved definitions and efficient constructions. CCS'06, Association for Computing Machinery, New York, USA, pp 79–88, Oct 2006. https://doi.org/10.1145/1180405.1180417
36. Song DX, Wagner D, Perrig A (2000) Practical techniques for searches on encrypted data. In: Proceeding of IEEE symposium on security and privacy. S&P. IEEE, Berkeley, USA, pp 44–55, May 2000. https://doi.org/10.1109/SECPRI.2000.848445
37. Kamara S, Papamanthou C (2013) Parallel and dynamic searchable symmetric encryption. In: Proceedings of the international conference on financial cryptography and data security. FC 2013, Springer, Okinawa, Japan, pp 258–274, Apr 2013. https://doi.org/10.1007/978-3-642-39884-1_22
38. Mohaisen SCKRCFL (ed) Security and privacy in communication networks. In: 15th EAI international conference. SecureComm 2019, vol 304. LNICST, Orlando, USA, Oct 2019
39. Bellare M, Boldyreva A, O'Neill A (2007) Deterministic and efficiently searchable encryption. In: Annual international cryptology conference. Springer, Santa Barbara, USA, pp 535–552, Aug 2007. https://doi.org/10.1007/978-3-540-74143-5_30
40. Van Liesdonk P, Sedghi S, Doumen J, Hartel P, Jonker W (2010) Computationally efficient searchable symmetric encryption. In: Proceedings of the 7th VLDB conference on secure data management. SDM'10, Springer-Verlag, Singapore, pp 87–100, Sept 2010. https://doi.org/10.1007/978-3-642-15546-8_7
41. Cash D, Jaeger J, Jarecki S, Jutla CS, Krawczyk H, Rosu MC, Steiner M (2014) Dynamic searchable encryption in very-large databases: data structures and implementation. In: Network and distributed system security symposium (NDSS'14), vol 14. Citeseer, San Diego, California, pp 23–26, Feb 2014. https://eprint.iacr.org/2014/853.pdf
42. Chen L, Lee WK, Chang CC, Choo KKR, Zhang N (2019) Blockchain based searchable encryption for electronic health record sharing. Future Gener Comput Syst 95:420–429. https://doi.org/10.1016/j.future.2019.01.018
43. Zhang R, Xue R, Liu L (2017) Searchable encryption for healthcare clouds: a survey. IEEE Trans Serv Comput 11(6):978–996. https://doi.org/10.1109/TSC.2017.2762296
44. Li H, Yang Y, Dai Y, Yu S, Xiang Y (2017) Achieving secure and efficient dynamic searchable symmetric encryption over medical cloud data. IEEE Trans Cloud Comput 8(2):484–494. https://doi.org/10.1109/TCC.2017.2769645
45. Whitmore A, Agarwal A, Da Xu L (2015) The Internet of things—a survey of topics and trends. Info Sys Front 17(2):261–274. https://doi.org/10.1007/s10796-014-9489-2
46. Atzori L, Iera A, Morabito G (2010) The internet of things: a survey. Comp Netw 54(15):2787–2805. https://doi.org/10.1016/j.comnet.2010.05.010
47. Buckl C, Sommer S, Scholz A, Knoll A, Kemper A, Heuer J, Schmitt A (2009) Services to the field: an approach for resource constrained sensor/actor networks. In: International conference on advanced information networking and applications workshops. IEEE, pp 476–481
48. Shamir A (1979) How to share a secret. Commun ACM 22(11):612–613. https://doi.org/10.1145/359168.359176

Dictionary Based Gender Identification and Gender Based Sentiment Analysis with Polarized Word2Vec

Navodita Saini and Dipti P. Rana

Abstract Each gender is having its special behaviour which can be reflected in every field of social media. During the pandemic of COVID-19, people used twitter to discuss the issues caused by COVID-19 disease. As Twitter does not disclose the gender of the user, in this study we have discussed different kinds of approaches used to identify the gender. From the literature review, it is found that the dictionary-based approaches are the best suitable approach when we are working with the sentiment analysis of unlabelled data. This study is about the analysis of ten kinds of emotions of males and females by which we can observe how they reacted in this pandemic. The research proposes a dictionary-based approach to identify the gender and then analyzed sentiments using the cluster-based approach is applied onto word vectors after multiplying them with sentence's polarity. The proposed approach is compared with the existing approaches with different data set and found that our proposed approach depicts good accuracy of sentiment analysis of unlabelled gendered data.

Keywords Sentiment analysis · COVID-19 · Dictionary · Word2Vec · Polarity

1 Introduction

Today's virtual technology has generated a large amount of data that might be used by various researchers to determine the sample based on content, emotions, deportment, network surfing, and other factors [1]. If you look at these patterns in depth, you'll notice that they have an effect on gender. Every gender has its own distinct personality and behaviour traits, which they may display through the use of language in situations like as offering an assessment, disseminating information, making friends, complying with topics/other consumers, and so on. All of this information is used by many groups

N. Saini (✉) · D. P. Rana
S. V. National Institute of Technology, Surat, Gujarat, India
e-mail: navoditasaini22@gmail.com

D. P. Rana
e-mail: dpr@coed.svnit.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_54

for their own benefit to encourage the goods in terms of recommendation of an item, link, video, music, and so on, without taking gender prejudices into account. The user's language's syntactic and semantic capabilities can reflect implicit biases in its creators' gender [2].

Throughout the world, the male-to-female ratio is unbalanced. Furthermore, this ratio is unbalanced on social media, as evidenced by registered consumers, customer status, customers participating in discourse, and many other factors. On social media networks, the female-to-male ratio may be extremely skewed. According to several studies, non-technical social media networks such as Snapchat, Pinterest, Instagram, and Facebook have more girls than boys, but technical and academic social media networks such as Twitter, LinkedIn, and others have less females than males. "Twitter", one of the most well-known social networking sites, shows that its members are discriminated against because of their gender.

Because this article is on the gender perspective, there are some data about Twitter users who are female [3]. According to Pew Research Center's advertising and marketing audience figures, the global gender split is 62% male and 38% female. Within the United States, the gender split is 50–50. Women are far more likely than men to be the most prolific tweeters among American adults: 65% of Twitter's most active users are females. On the other hand, women in the United States are only slightly more likely than men to have private Twitter accounts. The male-to-girl ratio in India's Twitter ad target market is 85%, putting the United States in third place. In the Philippines and India, women account for 48.04% of the population, compared to 51% in the United States. Males make up 96% of the population. However, 85% of Twitter users are men, whereas only 15% are women [4].

Gender-based research continues to be a fallacy in the field of machine learning studies. The researchers were drawn to the gender imbalance ratio because of its impact on the emotions of textual content facts. During a difficult time, such as the COVID-19 pandemic, specific social media structures were used by customers throughout the pandemic year 2019–2020 to discuss the difficulties caused by the radical virus ailment, such as its symptoms, mental health, fitness services, advice, recovery rate, hospitalization/death ratio, containment region, supply chain and so on. These long-form tweets covered a wide range of topics, including gender, career, age, tradition, and other unique human characteristics.

To yield the promising information outcome from the shared statistics via assessing the gender's impact requires gender identity, which is not to be had with the records we retrieved from the twitter. In different studies, researchers did it in different ways. Some of them identified the gender manually by first name; other one performed it by using an online available API. From the literature review, discovered the disadvantages of gender identification approaches and found that for the unlabelled data, dictionary-based approaches work well. Thus, the researchers propose the dictionary based approach for gender identification.

After identification of gender, the most frequent word used by male and female has been discussed. By those words, some important psychological observation we will discuss. In addition, how we can use that observation in commercial, political, marketing, etc., purposes will be discussed. For analysis of tweet sentiment, in this study, the sentiment will be described in a very ragged way. It is not only the positive and negative sentiment but here ten types of emotions will be discussed.

As we are working with unlabelled data then most of the sentiment analysis is done by dictionary method. However, we have another approach by which we can find the sentiment. That is a cluster-based method approach. In this study, we will propose one approach, which improves the accuracy of sentient analysis of unlabelled data, and give a better accuracy than existing approaches. After identification of gender and emotion of the user, there is a need to know how accurate our output is? For accuracy check, we first label our data with the same tool we used for sentiment analysis and applied our proposed approach to know the accuracy.

2 Literature Survey

In this study, for gender identification, an idea with a proper diagram has been given, which is a dictionary approach with the most frequent words used by males and females. After adding more words, we can get better accuracy. For sentiment, analysis ten types of emotion have been discussed which give some psychological observations about the thinking process of males and females. It will not be possible when we work with only positive and negative sentiment. Following previous research by Hu and Kearney [2] used customers' first names to deduce their gender data. Their approach is constant with human perceptions of gender [2]. Other research additionally took the tweet because the facts and gender are taken into consideration as well, as they worked on weather trade perspectives of women and men by tweet [5]. They used the Mozdeh API to determine the consumer's gender, which identifies the gender based on the consumer's first name [6]. On this look at, Garcia-Rudolph et al., Garcia they created a few lines of code that began with each Twitter account and was observed through profile calls, profile descriptions, profile images, and tweets. After that, they double-checked their final gender codes for each Twitter user [7]. Go et al., presented one of the first research on sentiment analysis of Twitter data in 2009. Their research is based on Twitter sentiment categorization utilizing distant supervision [8], which can be used on both supervised and semi-supervised data. There's also some work on sentiment analysis of Twitter data. They investigated robust sentiment identification from biassed and noisy data on Twitter [9]. Twitter was used as a corpus for sentiment analysis and opinion mining by Alexander Pak and Patrick Paroubek. They utilized positive emoticons like ":)" and ":-)" for positive tweets and negative emoticons like ":(" and ":-(" for bad tweets [10]. To do so, we use several vectorization approaches such as term frequency-inverse document frequency (TF-IDF) [11] and Word2Vec [12] to convert the words into vectors. The main focus of this research is on the Word2Vec vectorization technique. There are numerous efforts on Word2Vec that use classification techniques to improve accuracy. However, the goal of this research is to discover a vectorization technique that uses clusters and improves accuracy.

There is a few studies which makes use of tweet for sentiment analysis and take into account gender differences perspectives. Hu and Kearney worked on Computational textual content analysis of Gender variations in Political dialogue on Twitter. They determined that in comparison to men, women typically had a more potent sense of organization awareness and concord and showed a preference to sell their tweets at the same time as keeping off addressing different users in political discussions [2]. Politics has lengthy been regarded as a "guy's problem". As an end result, girls are often assumed to be much less fascinated and knowledgeable in politics [13]. Kim Holmberg worked on gender variations inside the weather trade conversation on Twitter and confirmed that the woman tweeters tend to reveal greater interest and perception inside the anthropogenic effect on climate trade and in the direction of campaigns and agencies involved within the debate, male tweeters are extra concerned with politics associated with climate trade and connect extra (for one motive or the other) with people who have an extra sceptic stance within the weather exchange debate [5]. Women were determined to be extra concerned about weather alternate than men and possessing more scientific information about the issue, within the context of American public [14]. The work of Park analyzed subjects by means of a grouping of semantically comparable words routinely by using the use of LDA from 10 million messages of fifty two thousand fb customers and recognized subjects related to girl users as friends, family, and social existence, whereas subjects related to male users as swearing, anger, dialogue of items in place of people, and the usage of argumentative language. From 15,000 Facebook users, the equal authors observed the language utilized by women turned into interpersonally warmer, extra compassionate, well mannered, while the language used by males became colder, extra antagonistic, and impersonal [15].

Because gender information cannot be directly obtained from Twitter data, the approach for identification of the user's gender applied by the researchers is different. As Twitter does not disclose the gender of their user, so identification of gender is a tedious task. Most of the researchers did the identification by the name. There are some challenges has been observed, when we go with name of the user. We have to find the first name because when we give the whole name to the tools which are available in R and Python language it gives "Unknown" as an output. Identifying the first name itself is difficult because the names of the users are ambiguous. There is no specific pattern in the name so that we can select that particular part of name. We have to go manually name by name to find the first name. But when we are working with millions of tweet it is difficult to go manually. The disadvantage of working with R and Python tools are: They used dictionary method to find the gender by first name. In their dictionary only United States (US) and United Kingdom (UK) names are there [16]. So it doesn't work for another country name ex: India, Pakistan, China, Japan, etc. As we studied in related survey that some of the researcher did the gender identification by first name manually. This kind of approach takes more time and effort. According to that research the accuracy is 70% [17]. The next approach which is used for gender identification is some online API. The accuracy of these APIs is more than 95%. It also works on dictionary method and here no need to give only the first name we can give the name as it is. In India, most of the APIs are banned. And from the available one you have to buy the package which depends on how many users gender yow want to find. Because it is illegal to find the customer or user gender if that platform itself wants to hide it.

3 Methods and Proposed Approach

3.1 Gender Identification

As twitter does not disclose the gender of the users, in the theoretical background it is already discussed that gender of the user is identified by the first name. It has been discussed that there are disadvantages when we go with first name gender identification approach. Here we are giving an approach in which the words used by men and women are used. This is a dictionary approach where we prepare a dictionary by following some steps and then use that dictionary for gender identification. Here.

In Fig. 1, it is explained that how we can make our own dictionary for gender identification. For this we have to take some pre-prepared data set label with gender. Then do the stemming and tokenization process. Separate the tweet for male and female. Then take word one by one from one gender and search it on the another gender, if that word is not present in another gender tweet then put that word in the dictionary of that gender from where they used. But if a word is used by both male and female take the average for both male and female and put that word in the maximum average gender dictionary. To use this dictionary for gender identification take the tweet or review of the user and apply pre-processing, stemming and tokenization then try to count the number of word matching with both dictionary and label that user's name with maximum count dictionary type.

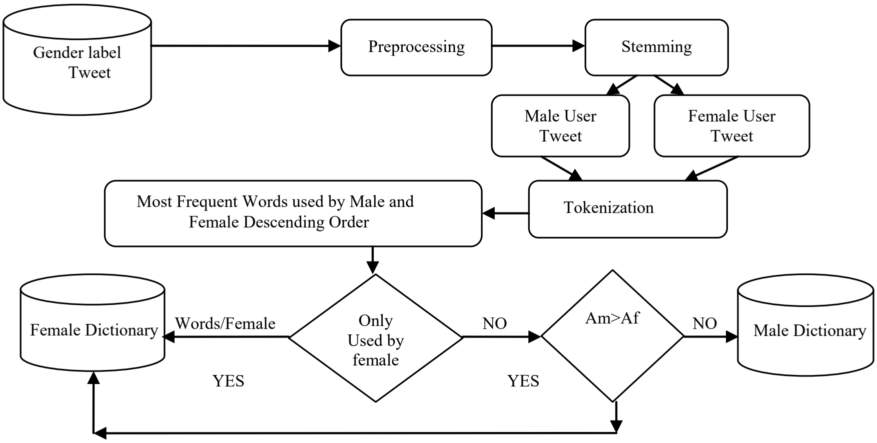


Fig. 1 Proposed dictionary

where:

Am = word Frequency/Total words used by male.

Af = Word Frequency/Total words used by female.

3.2 Sentiment Analysis

There are mainly three techniques for sentiment analysis for the Twitter data [13]. That is Supervised Learning Approaches, Lexicons-based Approaches and Cluster-based Approaches.

Supervised learning methods is primarily based on label dataset and accordingly the labels are supplied to the model throughout the system. Those label dataset are trained to get meaningful outputs while encountered at some point of selection making [18]. Lexicon-based approaches totally is based on the method that makes use of sentiment dictionary with opinion words and fit them with the records to decide polarity. They assign sentiment scores to the opinion phrases describing how positive, negative and objective the phrases contained inside the dictionary are. Lexicon-based methods especially rely upon a sentiment lexicon, i.e., a set of acknowledged and pre-compiled sentiment phrases, phrases or even idioms, evolved for classic genres of communication, which includes the sentiment annotation lexicon [18].

In this study, we are working with tweets which is unlabelled so we have to go with Lexicon-Based approach. We have used the tool name as get_nrc_sentiment() from library syuzhet. This tool gives ten emotions percentage as an output [19]. So now we are going to discuss how this tool works and what kind of dictionary used by this tool for annotation. The dataset that is used as a dictionary referred to as EmoLex, is commonly as large as the only other acknowledged emotion lexicon, WordNet having an effect on Lexicon. Greater importantly, the phrases in this lexicon are cautiously selected to encompass a number of the maximum common nouns, verbs, adjectives, and adverbs. Past unigrams, it has a large range of generally used bigrams additionally include a few words from the overall inquirer and a few from WordNet affect Lexicon, to permit comparison of annotations among the numerous sources. WordNet affect Lexicon has some hundred words annotated with the emotions they evoke. It became created via manually figuring out the emotions of some seed words and then marking all their WordNet synonyms as having the same emotion [20].

Cluster-based Approaches

The solution of previous explained problem is Cluster-based approach. When we work with clustering-based approach the main points to be noted that [17]. Clustering-related strategies can't generate a version inclusive of one trained in classification, which may be used to predicate the class of any new files. Clustering groups files into elements, but does not indicate which is positive and which is negative. The accuracy can every now and then be particularly low. 4. Clustering consequences are unstable because of the random selection of centroids in K-Means.

The number of steps require when we are working for sentiment analysis of any kind of reviews data and tweets data from cluster-based approach [17].

(1) Pre-process the data set. (2) Apply feature extraction algorithm for converting this into vector space. (3) Apply K-Means clustering algorithm. Clusters are primarily clustered into two clusters which are expected to be a positive group and a negative group.

Steps require in pre-processing of data is already discussed. For feature extraction and converting the document into vector space, some approaches are available, Bag-of-word, TF-IDF, Word2Vec. etc. Before applying these steps in unlabelled data we have to make sure that which features extraction technique gave us a more accurate output. Therefore, for finding that how much accurate these algorithms are, and how much effective this approaches are, we have to apply it in label data and try to find the accuracy. For accuracy check of the approach label data has been used so it is known to the researcher that in which sentiment how many reviews are there.

Gang Li and Fei Liu studied on this approach and worked on accuracy over movie review data set. They did multiple units of experiments with distinctive records sets, each with a size of 600 with a distribution of 300 positive/300 negative. No considerable distinction was observed among these experiments. Consequently, it is believed that the experimental result will now not be exceedingly affected by different selections of files. According to their experiment when Bag-of-Words has been taken for feature extraction and they used the output of vector in effective way not just as it is and consider this vector for K-Means clustering algorithm the average accuracy of sentiment is 55.7–57.7%. Then they did the same analysis with same data set but for feature extraction they took TF-IDF algorithm also here too they used that vector in a best way after applying some more approaches. The average accuracy of this approach was 72.2–73.1% [17].

In this study, one of the best approaches for feature extraction has been discussed. Here we have compared the accuracy for sentiment of the output of cluster made by Word2Vec and polarity then we discussed the proposed approach, which improves the accuracy. For implementation two types of data has been taken. Review data set of movie and tweet data set related to flight experiences. In movie data set, there are 1000 label reviews, which show the reaction of the people after watching that particular movie. In this data set 497 reviews are negative and 503 reviews are positive [21]. Flight related tweets are the feedback of the people who has taken that flight and experience the service. There are total 600 tweets we have taken in which 300 are positive and 300 are negative [16]. The reason behind taking these two different data sets are, generally review of movies that has so many sentences because there is no restriction of words but in tweet there are some limitations, we can only write 280 characters. These differences affect the accuracy as the reviews holds more words, the accuracy are less compared to tweet data set [17]. That is why for better visualization we have taken these two different kinds of data set. After selection of the data set do pre-processing then, train the model with Word2Vec algorithm, which gives a vector form of the words feature. Feature extraction technique used to hold the semantic value of the sentences. So after creating the model pass the sentences one by one and take the vector form of the sentences. In this study, we have taken average of the

Table 1 Accuracy with Word2Vec

Data set	Cluster	Total	Positive	Negative	Avg. accuracy (%)
Movie review data	1	754	386	368	52.22
	2	246	115	131	
Tweet data	1	258	77	181	67.68
	2	342	223	119	

Table 2 Accuracy by proposed work

Data set	Cluster	Total	Positive	Negative	Avg. accuracy (%)
Movie review	1	524	120	404	77.94
	2	476	101	375	
Tweet data	1	244	30	214	81.77
	2	356	270	86	

vector. Then apply K-Means clustering approach with centre value 2. In this study, we have taken average of the vector. Therefore, before giving our proposed approach, we have compared here the accuracy of some existing approaches so that we can see how our proposed works improve the accuracy. For comparison, we have taken the same data set, which explained earlier.

In Table 1, we have applied the existing approach, which is applying clustering on Word2Vec, and we can see here the accuracy. By this Table 2, we can see that the accuracy is very less when we apply K-Means on Word2Vec [12]. It is visible that the accuracy with Tweet data is more than review data. Accuracy of sentiment with polarity is 70% [22].

3.3 Proposed Approach

As the accuracy of present approaches has been explained, there is a need to know that do we improve the accuracy. In this study, we are giving one approach by which the accuracy of sentiment analysis by clustering method was improved. These are some of the steps we need to follow to achieve this accuracy.

(1) Obtain word vector of each sentences and take the average. (2) Obtain polarity of each sentence. (3) Multiply the polarity with word vector average. (4) Apply K-Means cluster algorithm on multiplied word vector value with polarity.

The purpose of taking Word2Vec vector form of the sentence is it holds the semantic value or meaning of the sentence. In the second step polarity of the sentence has been calculated which provide how positive or negative the sentence is, so that the sentiment is a very finer way can be calculated. As after knowing the semantic value and polarity of the sentence when we multiply, it gives an effective sentiment mathematical value. For testing to find how much effective this approach is the same data set has been taken. After taking the data set and following the explained steps, we got the following output.

In Table 2, we can see that with our proposed approach the accuracy was improved. The accuracy of our proposed approach with movie review data is 78% and for the tweet data, it is 82%. Because here in this approach, we used machine learning approach and dictionary approach. Word2Vec holds the semantic value of the sentence and polarity give the sentiment value of the sentence, and combination of this give us a better accuracy. By the same approach when we have a label, data instead of clustering approach we can use any classification algorithm and the accuracy of this classification model is 89% [23].

4 Implementation and Recommendation

4.1 Corpus of Tweet

The data set which has been used for this work is the tweet data set with keyword “corona”, “#corona”, “corona virus”, “coronavirus”, “covid”, “#covid”, “covid19”, “#covid19”, “covid-19”, “#covid-19” etc. In this study, we used pre-prepared COVID-19 tweet data. In this data, there are 13 columns with name “User_name”, “User_location”, “User_discription”, “User_created”, “User_Followers”, “User_friends”, “User_favourite”. “User_verified”, “Date”, “Text” and total 179,105 tweets are there in “Text” column [22]. We can see in the data that there is no column with gender.

First, we have to do pre-processing then to find the root word stemming is required. To make the sentence into words we have to go to tokenization. After that pass, that words to both the dictionary and count the number of words matching separately. Then compare the frequency of words and add that user name with maximum word count dictionary value. After adding more words, the accuracy of the dictionary can be improved. As in this study, the data set is related to covid-19 in which there are so many irregular used words are there. Therefore, there is no pre-prepared data set found where the gender has been given. Therefore, for identification of the gender of the user with covid-19 tweet is difficult here. However, we can use this kind of dictionary for gender identification of regular tweet users. For this study gender identification has also been done by online available API, the accuracy of this API is more than 95% [24].

After identification of gender, we observed most frequently used words by male and female. That shows the behaviour and we can conclude some psychological observations.

In Figs. 2 and 3, we can see that the way of expressing the view of this pandemic of the male and female are very different. Following are some observations from this graph: Men addressing this virus directly but women are more interested in

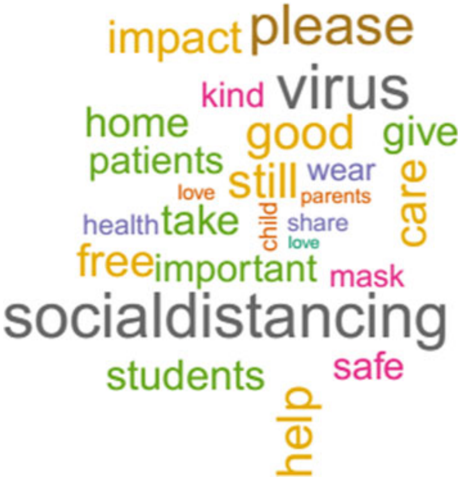


Fig. 2 Most frequent word used by male

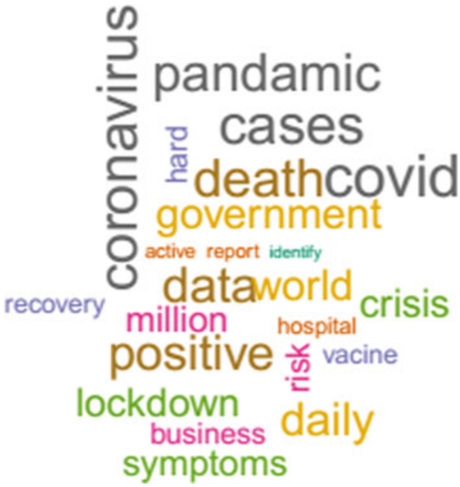


Fig. 3 Most frequent word used by female

saying about something without addressing it directly by their name. We can use this observation in many fields. Male user are more interested to know that the overall count of cases in world, number of positive cases in a day, number of death in a day due to covid. However, female is more interested about precaution. Female are worried about wearing a mask, following the social distancing and how to stay safe. Male are talking about the government policies, data provided by government, and more worried about the vaccine. Men are worried about business but women are thinking about the children, home and students. Women talking about love care help and safety, which shows their positivity and strength in any difficult condition.

After identification of gender of each tweet pre-processing of tweet has been done that includes, Removal of Punctuation, Remove Number Remove Stop words, Remove URL Remove Special Characters. Because it has been referred to earlier that there are, 179,105 instances of duration 25-07-2020 to 30–08-2020 has been taken for this study. On this, 10.33% of tweets, i.e. 18,502 instances are from the woman gender, 24.21% tweets i.e. 43,373 instances are from male gender and ultimate all are from the business enterprise, news channel, etc. Therefore, now to carry out the calculation of emotions sampled instances one by one for each genders and kept one sample of all the instances together. However, this studies primary attention is at the emotions of genders most effective. As it has been explained in chapter, method that there are ten emotions we are going to compare with both the gender. The tool which is used for this work is get_nrc_sentiment() from library syuzhet [19].

4.2 Result Discussion

The subsequent figure represents the emotions percentage of the tweet that is published by (a) male user, (b) female user, and (c) combined genders and different business enterprise, news channels, and so on. We will see the variations of those three categories that are very marginal. However, this studies most important cognizance is gender-wise emotion analysis. Right here the instances of the female are most effective 10.33% tweets i.e. 18,502 tweets and 24.2% tweets i.e. 43,373 tweets are of the male. The following graph depicts the percentage of tweets with positive feelings published by (a) male users, (b) female users, and (c) mixed genders and various business enterprises, news outlets and so on. We'll see variations of those three groups that are really minor. The gender-based emotion analysis is, however, the study's most essential finding. Female instances are most effective in this situation. Males account for 10.33% of tweets (18,502 tweets) and 24.2% of tweets (43,373 tweets). Figure 4 depicts the percentage of emotion for each gender using the same number of tweets (18,502) from both tweeter users.

Figures 4, 5, and 6 show how the epidemic has shown all of the different types of emotions in variable percentages, as well as the insights and strengths of Twitter users at this tough period. Fear was the dominant negative feeling, owing to the high fatality rate. Following terror, the second most negative sensation became anticipation as people discussed how to stop the sickness from spreading and how to relax during this difficult period. The third negative emotion is sadness, which is more prevalent in women than in males. In challenging times, both genders display more positive emotion than negative emotion, according to the derived emotion. Nonetheless, the pandemic has shown nearly equal percentages of negative emotion fear of disease and positive emotion trust, but these two emotions had a higher percentage than all of the negative and positive sided emotions, demonstrating the pandemic's havoc while also demonstrating the strength to face it. Furthermore, such strength can be demonstrated in the percentage of people who express joy. Despite the challenging circumstances, people attempted to retain a positive attitude toward their own family, and as a result, an increase in the emotion of trust may be detected. As a result, the gender perspective of tweets and other gender-based evaluations of characteristics gave

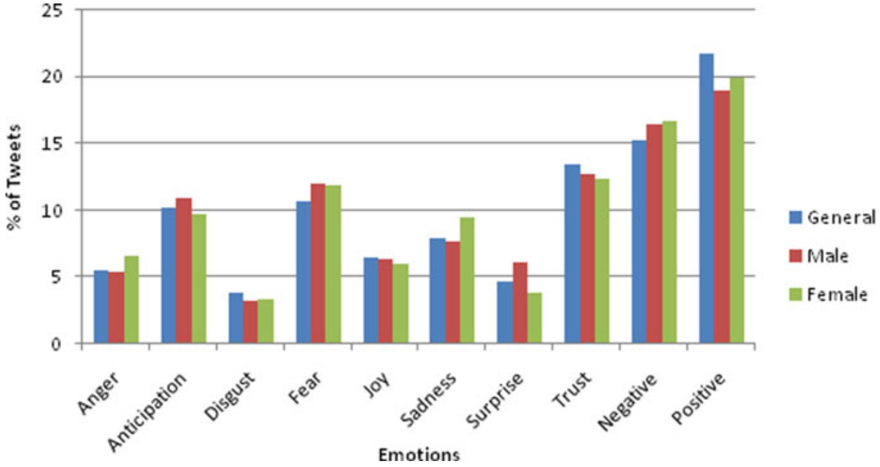


Fig. 4 Percentage of emotions

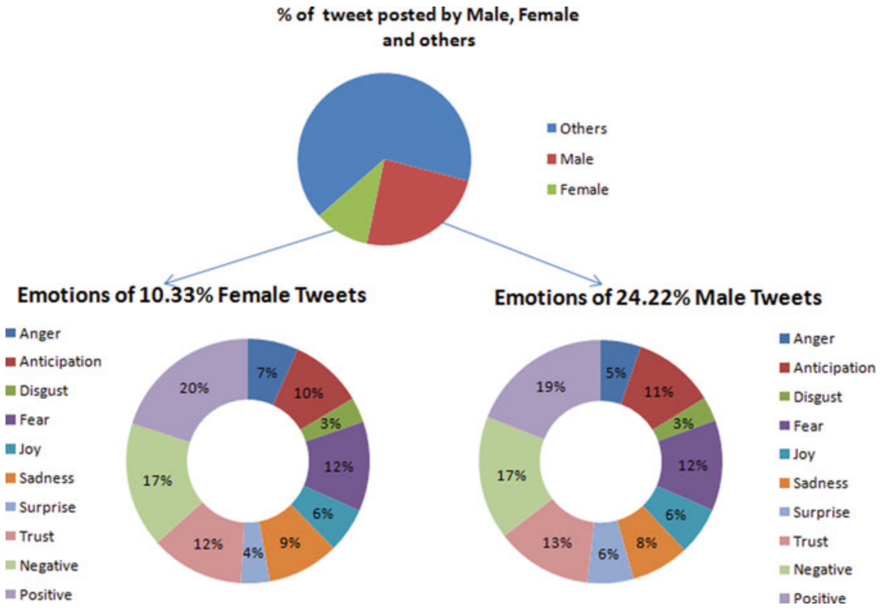
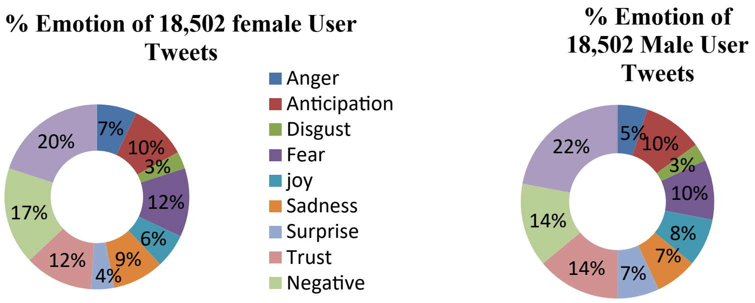


Fig. 5 Categorized percentage of tweets and gender-wise percentage of emotions

insightful information, emphasizing that this gender perspective of tweets and other gender-based evaluations of features can provide more intuitive real-life applications.



% Emotion of 18,502 female User Tweets

% Emotion of 18,502 Male User Tweets

Fig. 6 Categorized percentage of emotions with the same number of tweet by female and male

4.3 Accuracy of Sentiment Value with Gender

For gender analysis, we have used an API and the accuracy of API is more than 95%. For sentiment analysis, our aim of this study is to observe ten kinds of emotion. For this, the tool we have used is a dictionary approach, which has been already discussed. We have proposed one cluster based sentiment analysis and saw their accuracy with two different kinds of data set. So to know how much the dictionary based approach and the gender identification is effective or accurate we took this tweet data set, applied our proposed approach, and got that the accuracy. We did not separate the gender for male and female.

We took that combined data followed the same step and saw in which cluster how many male and female are there as we have already gender label tweet. Then we label each sentence with their emotion with the same tool we used for sentiment analysis. For best-suited number of cluster there is a tool named, as "Nbclust" are available, which analyzes all the combination, observe the output, and give us the best-suited number of cluster. For our data they gave, best-suited number of cluster is three. Therefore, we made three clusters, observe the cluster, and counted that for male and female how many number of tweets present on that particular cluster and for which emotion that particular cluster representing for. From Table 3, we can see that the accuracy of sentiment analysis from gender perspective output by our proposed approach is 87%.

Table 3 Accuracy of sentiment with gender

Cluster	Emotion	Proposed %		Actual %	
		Male	Female	Male	Female
1	Sadness, anticipation, fear	37	37	41	42
2	Positive, trust, joy, surprise	29	32	34	35
3	Anger, negative, disgust	21	18	25	23
Avg. accuracy (%)	87				

5 Conclusion

Gender bias data can be used to create new quantitative and qualitative research for a variety of applications. This study looked at the gender perspective of tweet data and identified a number of difficult situations that could help to improve the application of real-world effectiveness to devise a novel approach of gender-based evaluation to perceive and dismiss statistics that affect gender-based emotion participation in social communities. The challenges of gender identification are also examined in this paper, along with a recommended dictionary technique for gender identification. The accuracy of the methodologies available for unlabelled data sentiment analysis was also addressed in this study, as well as one recommended approach to improve the accuracy of sentiment analysis with unlabelled data. This study observed the thought of male and female about the COVID-19 pandemic.

6 Future Work

As Twitter is the best social media platform to analyze emotion of the user by their tweets. Researcher use their tweets for different perspective. However, Twitter does not disclose the gender of the user. There is an improvement scope that is open for the gender perspective research to improve the accuracy of the model with the help of specialized symbols used in the text with intelligent machine learning algorithms.

References

1. Gaind B, Syal V, Padgalwar S (2019) Emotion detection and analysis on social media. arXiv preprint arXiv:1901.08458
2. Hu L, Kearney MW (2020) Gendered tweets: Computational text analysis of gender differences in political discussion on twitter. J Lang Soc Psyc, 0261927X20969752
3. Sehl K (2020) Top twitter demographics that matter to social media marketers. https://blog.hootsuite.com/twitter-demographics/, May
4. M. of Statistics and P. Implementation (2020) Ministry of statistics and program implementation. http://mospi.nic.in/sites/default/files/reportsandpublication/statisticalpublication/social statis-tics/WM17Chapter3.pdf, Dec
5. Holmberg K, Hellsten I (2015) Gender differences in the climate change communication on twitter. Internet Research
6. Mozdeh (2013) http://mozdeh.wlv.ac.uk/, Apr
7. Garcia-Rudolph A, Laxe S, Saurí J, Guitart MB (2019) Stroke survivors on twitter: sentiment and topic analysis from a gender perspective. J Med Internet Res 21(8):e14077
8. Go A, Bhayani R, Huang L (2009) Twitter sentiment classification using distant supervision. In: CS224N project report, Stanford, vol 1, no 12
9. Barbosa L, Feng J (2010) Robust sentiment detection on twitter from biased and noisy data, in Coling. Posters 2010:36–44
10. Pak A, Paroubek P (2010) Twitter as a corpus for sentiment analysis and opinion mining. LREc 10(2010):1320–1326

11. Rajaraman A, Ullman JD (2011) Mining of massive datasets. Cambridge University Press
12. Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781
13. Campbell R (2012) What do we really know about women voters? gender, elections and public opinion. The Political Quarterly 83(4):703–710
14. McCright AM (2010) The effects of gender on climate change knowledge and concern in the American public. Popul Environ 32(1):66–87
15. Park G, Yaden DB, Schwartz HA, Kern ML, Eichstaedt JC, Kosinski M, Stillwell D, Ungar LH, Seligman ME (2016) Women are warmer but no less assertive than men: Gender and language on facebook. PLoS ONE 11(5):e0155885
16. I. Saeta (2016) gender-guesser. https://pypi.org/project/gender-guesser/, Dec
17. Li G, Liu F (2012) Application of a clustering method on sentiment analysis. J Inf Sci 38(2):127–139
18. Kharde V, Sonawane P, et al (2016) Sentiment analysis of twitter data: a survey of techniques. arXiv preprint arXiv:1601.06971
19. https://www.rdocumentation.org/packages/syuzhet/versions/1.0.6
20. Strapparava C, Valitutti A, et al (2004) Wordnet affect: an affective extension of wordnet, 4(1083–1086):40
21. Kaggle (2020) Data set. https://storage.googleapis.com/kaggle-data-sets/203733/447609/bundle/archive.zip?, Dec
22. Kaggle (2021) Data set. https://storage.googleapis.com/kaggle-data-sets/17/742210/bundle/archive.zip?, Jan
23. Parikh Y, Palusa A, Kasthuri S, Mehta R, Rana D (2018) Efficient Word2Vec vectors for sentiment analysis to improve commercial movie success, pp 269–279
24. Gender API (2021) Gender API. https://gender-api.com/en/account/overview, Jan

Exploratory Data Analytics of Total Population Over Fertility Rate in South Korea

Anusha Ganesan and Anand Paul

Abstract The Republic of Korea is experiencing a demographic crisis with low birth rates and aging population. As the present circumstance represents a developing danger to the supportability of its economy, schooling, accounts, and protection, there is a critical requirement for definite and comprehensive activity. South Korea is one among the world's quickest maturing nations. There might be a crucial impact forecasted if the country does not mitigate this growing jeopardy on the population. In this paper, we designed a prediction model using the machine learning algorithm such as multiple linear regression on the total population and fertility rate data to do the exploratory data analysis about the future trend in the population and its impacts which would affect the wellfare of the nation. The correlation and prediction results showed an accuracy of 98%, with a declining trend of the fertility rate for the upcoming years as well. We evaluated the prediction model performance using root mean squared error and mean absolute error values in the training and testing of the model. Therefore, we concluded the paper with the future challenges for the country having such population trends.

Keywords South Korea · Total population · Younger population · Fertility rate · Multiple linear regression · Machine learning

1 Introduction

South Korea began to have an older community in 2000 with the portion of older people being 7.2% of the entire inhabitants. By the end of 2008, the amount of people aged 65 or above surpassed 5 million, which roughly exceeded 10% [1] of the whole population. Then, it had an elderly society in 2018 with the amount of older people in age 65 and beyond contributed to 14% of the whole population. In 2020, the aging population accounted for roughly 15.6% of the whole population in

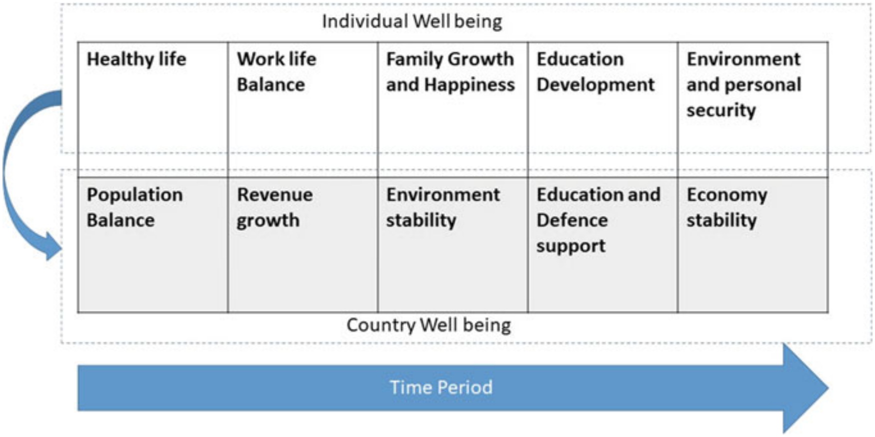
A. Ganesan · A. Paul (✉)
School of Computer Science and Engineering, Kyungpook National University, Buk-gu, Daegu, South Korea
e-mail: paul.editor@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*, Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_55

South Korea. South Korea is likely to face an intense demographic revolution, and it will develop into a super-aged civilization in 2026 with the number of senior citizens accounting for 20% of its population. Relating to the period taken by other countries to move from an aged to super-aged society, South Korea is expected to reach this transition by only 26 years, whereas France took 115 years; Britain required 91 years; 88 years for the USA; 36 years for Japan. Therefore, South Korea's aging inhabitants would contribute to 37.3% of its population in 2050, turning out to be the elderly populated country on earth, proceeded by Japan with 36.5%, Germany with 27.9%, Sweden with 27.1%, and France with 26.4% [1]. Moreover, a statistical study made by Statistics Korea also showed that Korea will be the world's most aged nation by 2067 with 46.5% of the entire population due to low childbirth and dropping marriage rates [2]. This trend in the population results in the demographic crisis which the country is facing. And, if it continues the future impacts for the nation reaches an overall state such as extinct state for the country by 2750 which is well-known for the technological advancements and safety. Apart from this long-term impact, there are some short-term impacts or immediate impacts that South Korea would have to face thereby affecting the people and economy of the country in the coming years. One of the major impacts is the healthcare needs for the elderly people. Due to the aging factor, there may be many unmet needs for the elderly people when compared to the younger population, as the healthcare system must be organized to support them from preventive, rehabilitative services that cover chronic diseases and promotion of life-long healthcare services. To have these healthcare support services up-running smoothly for the elder people, there must be a good revenue flow in the country which is mainly contributed by the workforce section of the country. On the decreasing trend for younger people, there would a less workforce population and it will be creating a huge impact on the country's economic status. On the other hand, South Korea is a country which is an inspiration for its growth, safety and its technological development [3]. Therefore, the government of South Korea has implemented some policies in place to increase the fertility rate of the country. Yet, the effectiveness of such policies and processes implemented should be measured to enrich the benefits of those processes. Thus, this paper deals with an exploratory data analysis for prediction about the fertility rate trend using multiple linear regression and we have studied the correlation between all the independent variables and dependent variable and, we evaluated the prediction model performance in terms of root mean squared error (RSME) and mean absolute error (MAE) values observed in the training and testing of the model. We also discussed the impacts which the country would face if the trend continued. Also, our results display that the decreasing trend of the fertility rate that can be stabilized if the respective policies and processes are being monitored periodically. The paper is organized in the below sections followed by the Introduction, Sect. 2 Literature Review, Sect. 3 Data Analysis, Sect. 4 Results and Discussion, Sect. 5 Conclusion.

2 Literature Review

This paper begins with a review of the literature regarding the population trend in South Korea since the 1960s. There was a study that explored the social and economic impacts concerning the existing situation of Korea's elderly population in 1996.They have concluded that the rapid increase of the aging population by then itself has brought an increase in the dependency ratio, increase an imbalanced sex ratio [4]. Also, the analysis made by the researchers in the article has shown that the urbanization of the elderly people lagged behind that of the entire population. The authors have brought out some information about the problem of the aging population being increased during 1996 [5]. Authors have suggested that it has been a multifaceted problem that required the state or country to have it dealt with and develop to ensure a suitable social welfare structure in place of a solution [6]. Another journal for social welfare has emphasized the social problems and recommended some solutions for them [7]. In the year 2005, an article explained theoretically the rapid decline in the fertility rate. They have also mentioned the causes of fertility decline such as labor market insecurity, gender fairness alignment, interruption in the initial family formation, and timing of first birth [8]. Owing to this, increase in the aging population and declining fertility rate, the government has made an effort for the elderly people to broaden the exiting National Pension scheme with the total benefits paid by this scheme in 2005 were 1.7 percent of the worker's taxable income. So, it is expected that the cost would be more than the contribution rate in 2025 and will continue to increase to 30% in 2050 [9]. In 2011, there is a literature work which has explained the issues and challenges that South Korea was facing at that moment due to the aging population [10]. According to an article in 2016, South Korea is having a good financial position for promoting the total fertility rate (TFR) which would not only provide childcare abilities and services, but also manage the economic and financial impacts as by 2030 over 1/3 of the entire population is expected to be senior citizens [11, 12]. As explained, mostly the researches made about South Korea's fertility problems were theoretical. Hence, while referring to some literatures about the fertility trend analysis and visualization, the regression method was found to be more effective in the prediction [13–15]. So, we have used multiple linear regression for the EDA to extract more accurate information and prediction result. Thus, we have focused this paper to show the statistical learning of the fertility rate along with the population data. Figure 1 shows a social model to explain what are factors that accounts for country's well-being based on the literature works and our understanding about the current situation of the country. The model represents that the individual well-being and country well-being are dependent on each other.



Individual Well being

Healthy life	Work life Balance	Family Growth and Happiness	Education Development	Environment and personal security
Population Balance	Revenue growth	Environment stability	Education and Defence support	Economy stability

Country Well being

Fig. 1 Social model for a country's welfare

3 Data Analysis

Using the open-source datasets [2, 16], we gathered the data about the total population, male population, female population,15–64 years aged female population, the fertility rate in South Korea till 2020.We performed exploratory data analysis (EDA) for this dataset. It is an approach for data analysis which employs various techniques such as graphical methods to maximize insight about the dataset, extract important variables, find out the outliers, test the underlying assumptions, and create a resource effective model. However, EDA is not identical to statistical graphics method although it uses a collection of techniques. It is an efficient method to reveal the underlying structure of the data and how to dissect the data and interpret the data. The following processes were carried out during EDA to get accurate and efficient prediction results.

3.1 Dataset Description

The dataset was integrated from the open-source datasets available from World Bank open data and Korean Statistics websites. The datasets extracted from these websites were having 265 rows and 65 columns, 325 rows and 62 columns, and 523 rows and 62 columns, respectively. The features of the dataset include year, total population, male population, female population, younger male population, younger female population, child population, and fertility rate in every city across the country.

3.2 Data Transformation

This is an initial and crucial step in data analysis as the learning outcome can become inappropriate if this step is not performed well. Hence, we executed this process in simple 4 steps to ensure the effectiveness of the overall process.

Data Interpretation

All the necessary data was not available in a single dataset from the open-source platform, therefore multiple datasets was fetched and analyzed to consolidate the required dataset. As a result of several data files involved, we had to define a specific data format to align the entire gathered data.

Pre-data Transformation Quality Check

After defining the data format for the complete dataset, pre-transformation quality checking of the data was performed to ensure that there are no missing or corrupted values from the source data that may create problems in the further steps of data transformation.

Data Transformation

Upon ensuring the quality of the data in terms of missing or corrupted values, we performed the data format transformation to convert the source data in the desired format. We have used data integration method for this process. This was a crucial step as we required to combine data residing from different sources which will provide a unified view of the data. There are two major approaches in data integration: Tight coupling and loose coupling approach. We followed the tight coupling approach as the data was pulled over from different sources into a single dataset, through extraction, transformation, and loading. It helped to provide a uniform data for advanced data summarization.

Post-data Transformation Quality check

Despite the pre-data transformation quality check, we also did a quality checking of the data after the data transformation as well, so that there could be no problems about data values in the file in the further stages of the analysis.

3.3 Data Cleaning

We worked on the data cleaning process for the dataset in 4 steps as follows:

Removing Duplicate or Irrelevant Observations

Identified the duplicate entries of the data and removed them. Also, we found some irrelevant entries and figured out the relevant values using the data scraping process.

Fixing Structural Faults

We experienced the faults such as boolean value, the precision value on the data for the population. Thus, these incorrect naming data, values, and typological faults were noticed and analyzed to categorize and enter the exact values and data.

Filtering Unnecessary Outliers

There were some one-off observations such as sudden spiking values on the datasets which were not having any appropriate reason supporting them. Hence, we had discovered those values recorded on the dataset were false and we found the correct values to ensure that the data was fit for further analysis.

Handling Missing Data

We had missing data like the dataset was not having values for a certain time, we gathered these values from other authentic sources to fill in these missing values.

Finally, we completed this process of data cleaning by cross validating the data to assure the data quality is maintained. We used Korean's Government statistical reports published in their authenticated websites for data cross-validation. As a result of the data cross-validation process required changes to the data were made.

3.4 Data Mining

Data mining is the process that satisfies the necessity of actual, manageable, and adaptable data analysis. It is an iterative process where the mining process can be developed, and some more data can be incorporated to get several efficient outcomes.

In this process, we started with the feature selection from the integrated dataset. The integrated dataset had a lot of features which were irrelevant for our prediction model. For instance, other countries, school going children, and many more. So, we dropped these irrelevant features to select the required features for our model. For checking and identifying the relevant feature in the dataset, we performed the exploratory data analysis.

Exploratory Data Analysis

As mentioned above in this process, we finalized that final features which are required for the final dataset so that our model is well trained and tested without overfitting and underfitting. Firstly, we performed the univariate analysis of the features that we thought to be important and relevant. Figure 2 shows the box plot for the features such as total population, year, younger male population, younger female population, and combined with fertility rate.

Then, we used scatterplot to do the bivariate analysis and understand the correlation among these features of the dataset. Figures 3, 4, 5, 6 shows the scatter plot for total population against the fertility rate, younger male population against fertility rate, younger female population against fertility rate, fertility rate with year. There

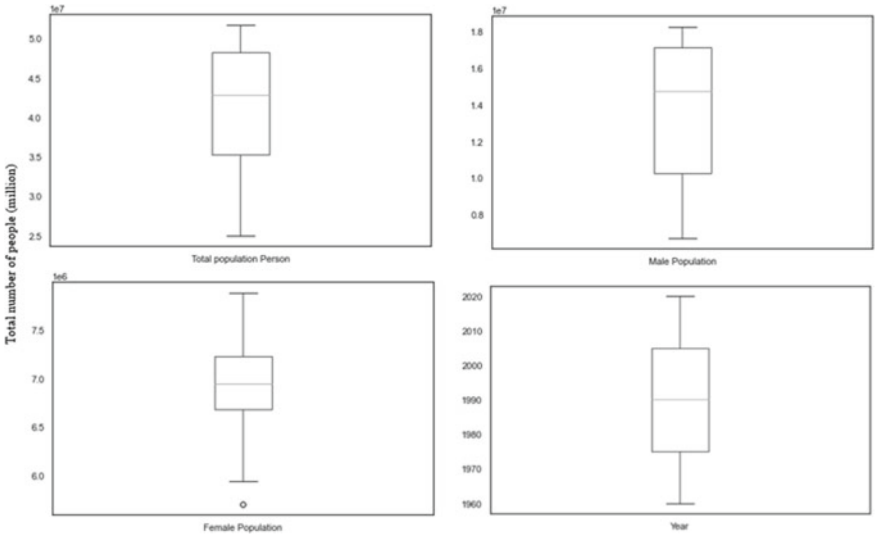


Fig. 2 Box plot for univariate analysis

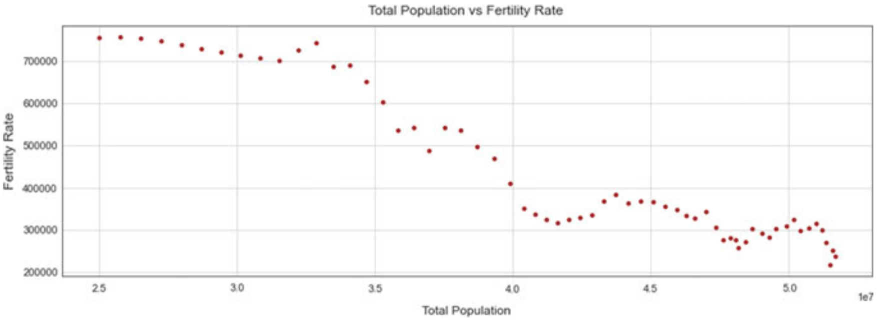


Fig. 3 Scatter plot for total population against the fertility rate

was negative correlation observed in these plots, confirming the declining fertility rate.

3.5 Data Representation

The data used for statistical analysis in the data mining process was visualized using the histogram to display the skewness of the data. Figure 7 shows the histogram representation of the independent and dependent variable from the input data.

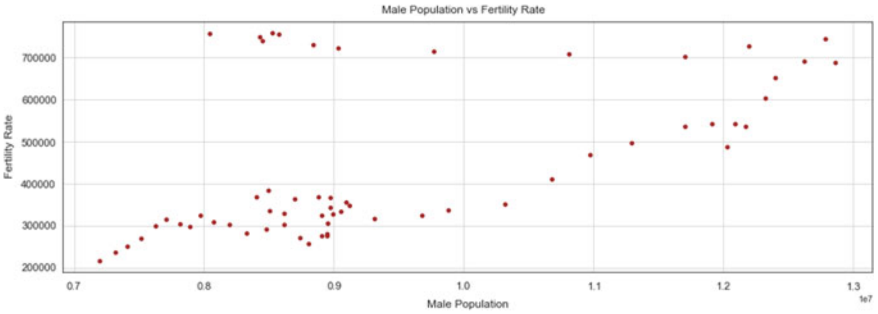


Fig. 4 Scatter plot for younger male population against the fertility rate

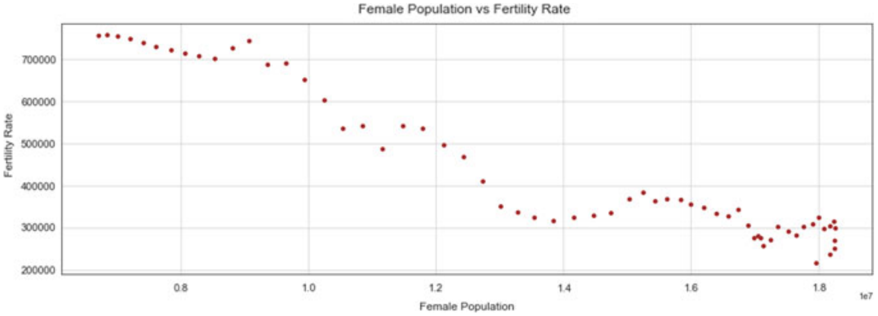


Fig. 5 Scatter plot for younger female population against the fertility rate

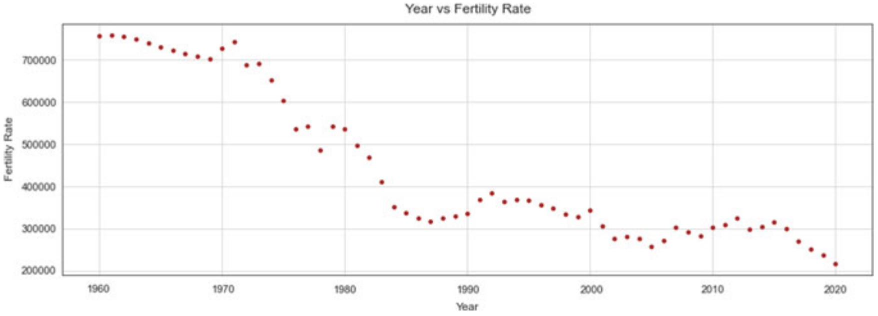


Fig. 6 Scatter plot for fertility rate against the year

3.6 Data Prediction

We have used the multiple linear regression algorithm to perform prediction using the exploratory data analysis from the preprocessed dataset. Because of the nature of the data, we decided to perform the regression analysis for the prediction model.

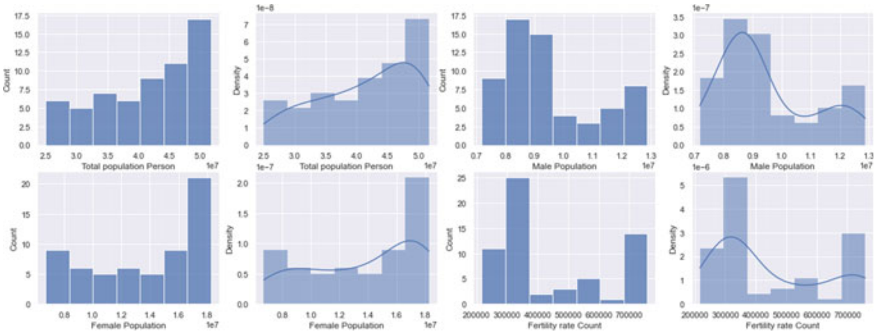


Fig. 7 Histogram of independent variables and dependent variables

Regression analysis is a set of statistical techniques that helps to estimate the relationship between a dependent variable for one or more independent variables. Linear regression makes a model for the relationship among two variables by fitting a linear equation to the recorded data, where one variable is an independent variable and the other is a dependent variable. However, for the integrated dataset, the linear regression was not able to fit directly as there was a hindrance in comparing the features of the dataset. So, we had to apply the normalization factor using the below formula to fit the data in the range from 0 to 1 and used multiple linear regression as our dataset involved analyzing more than one independent variable. The normalization technique supports to bring the entire set of probability distribution into the normalization alignment for better understanding of the relationship between the variables.

$$X\_\text{normalized} = (X - X\_\text{min}) / (X\_\text{max} - X\_\text{min}) \tag{1}$$

where X_max and X_min are maximum and minimum values in the dataset, X is the variable. Once we normalized the data, we began to model the data using multiple linear regression.

Multiple Linear Regression

Multiple linear regression (MLR) is a statistical method that can use several variables to predict the outcome of a different variable. The aim of MLR is to model the linear relationship between the independent variables and the dependent variable.

The equation of multiple linear regression is given by

$$y = b_0 + b_1x_1 + b_2x_2 + \ldots + b_nx_n + \varepsilon \tag{2}$$

y is the dependent variable and x_1, x_2, …, x_n are the independent variables and ε is the model deviation. Regression line parameters are b_0, b_1, b_2, b_3 …, b_n.

Therefore, we used this technique to study the correlation between the independent variables and the dependent variable from the dataset. Fertility rate is the dependent variable for designing our model and total population, younger male population, younger female population are the independent variables. The entire dataset was split into the ratio of 80:20 for training and testing the model. The performance of this designed model was evaluated using the error observed in the training and testing process.

4 Results and Discussion

Multiple linear regression depicted a negative trend on the dataset for the fertility rate in the next 10 years. Figure 8 explains that there is an upper confidence bound which shows that the fertility rate can be gradually made to move towards the positive trend and thereby improving the population. And, there is lower confidence bound explaining that there are much negative chances of the fertility rate to drop down.

MLR technique helped to visualize the correlation of the fertility rate against the total population, younger female, and male population. There was a declining negative correlation observed on the fertility rate against all the independent variables. The training and testing loss observed in the model confirmed the goodness of the prediction model. Figure 9 shows the prediction and observed value of fertility rate in training phase.

Figure 10 shows the prediction and observed value of fertility rate in testing phase. Since there was slight difference in error observed between the training and testing process, we performed the k-fold cross-validation for the prediction model. As a result of the cross-validation, the prediction accuracy of 98% was confirmed. The RMSE and MAE values observed in the training and testing phases proves the efficiency of the model. These values are recorded in Table 1. And, this shows that

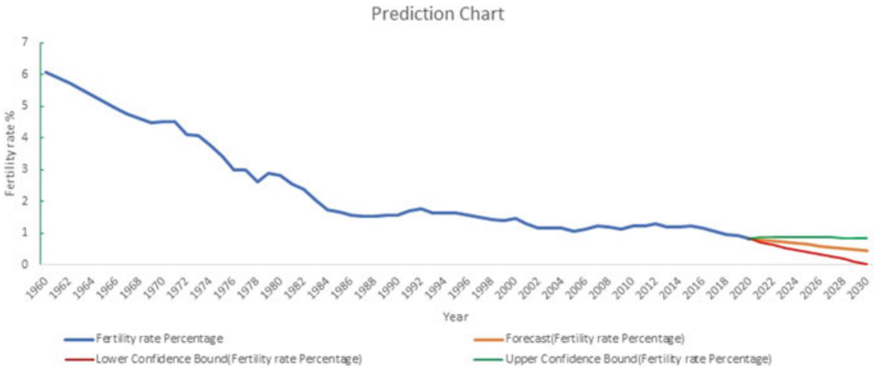


Fig. 8 Prediction chart for fertility rate trend in South Korea

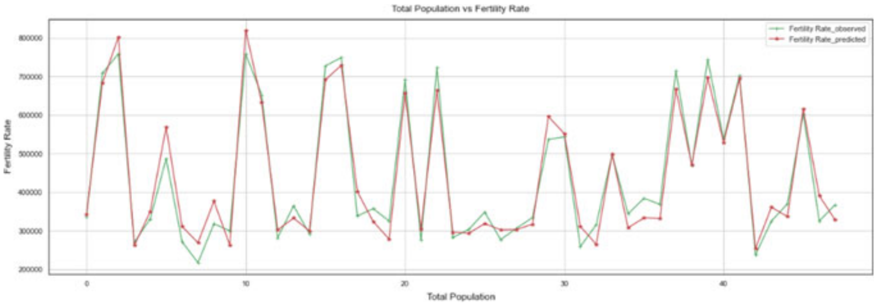


Fig. 9 Fertility rate prediction in Training phase

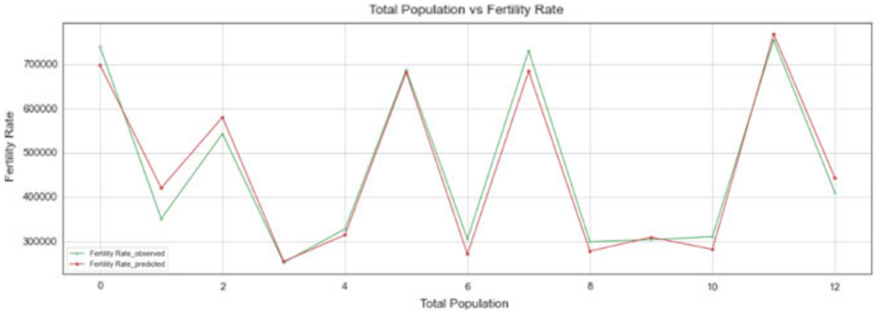


Fig. 10 Fertility rate prediction in testing phase

Table 1 Error values observed in training and testing process

Error	Training	Testing
RMSE	37,662.74	32,768.44
MAE	32,029.36	27,023.99

the error difference between the training and testing process was very negligible and therefore the best accuracy of the model was achieved.

From the experimental results, we understood that unless there is a monitoring process followed by the government to ensure the effectiveness of the processes implemented to promote the fertility rate, the country is likely to reach a state of non-existence in the future years. Figure 11 shows the cause-and-effect diagram for the clear understanding about the causes which can give raise to such an effect of South Korea might becoming extinct.

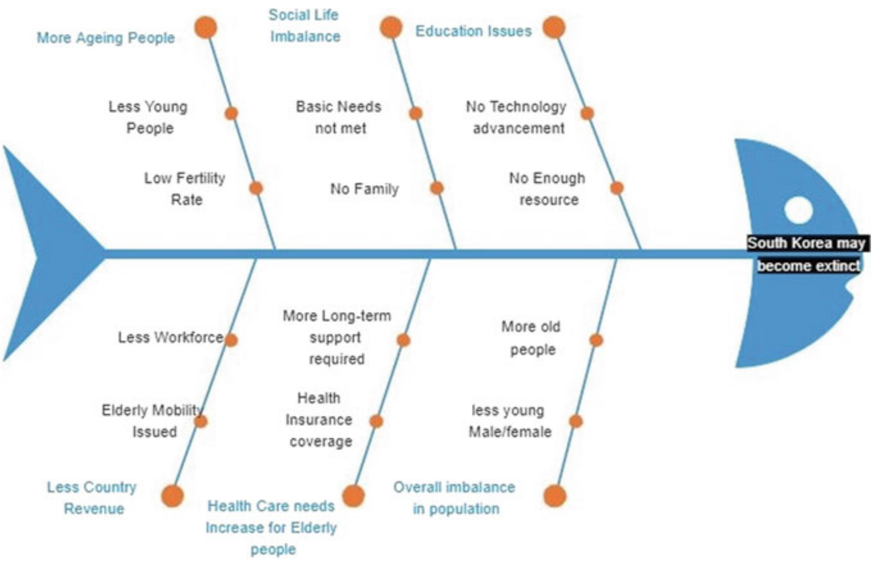


Fig. 11 Cause and effect diagram

5 Conclusion

South Korea is one of the most world's innovative nations and remarkable for its outstanding performance in research and development intensity, to become a global leader in the information and communication technologies and has emerged from a historically top-down innovation system. Being such a pride country, it is facing the fertility rate (births per woman) going on decreasing trend. This is very likely to impact on the existence of the country. This trend can remarkably make an effect for the country financial growth and economic status. Hence, this paper concludes with the awareness about the effects for such trend in the population. Every citizen and the government must take responsibility to contribute toward changing trend on the population and fertility rate in positive way to mitigate the big threat which is gradually following South Korea. Once, the country has achieved by overcoming the big threat of becoming extinct; South Korea would continue to be the global leader in the field of research and development with their innovative minds and thoughts which benefit not only to their nation but also to the entire world.

Acknowledgements This study was supported by the BK21 FOUR project (AI-driven Convergence Software Education Research Program) funded by the Ministry of Education, School of Computer Science and Engineering, Kyungpook National University, Korea (4199990214394). This work was also supported by National Research Foundation of Korea, (NRF-2020R1A2C1012196).

References

1. Hyung PH (2009) South Korea looming age crisis. Japan Spotlight, 26–27
2. kostat. http://kostat.go.kr/portal/eng/pressReleases/11/3/index.board
3. Park GR, Seo BK (2021) Mental health among the Korean older population: how is it related to asset-based welfare? J Appl Gerontol 40:142–151. https://doi.org/10.1177/0733464820917295
4. Kim IK, Liang J, Rhee KO, Kim CS (1996) Population aging in Korea: changes since the 1960s. J Cross Cult Gerontol 11:369–388. https://doi.org/10.1007/BF00115802
5. Development S (2017) Aging and social policy in Korea. Authors: Sung-Jae Choi (1996) Source : Korea J Popul Develop 25(1):1–25, July. Published by: Institute for Social Development and Policy Research (ISDPR) Stable URL, 25:1–25
6. Kim KW, Kim OS (2020) Super aging in South Korea unstoppable but mitigatable: a sub-national scale population projection for best policy planning. Spat Demogr 8:155–173. https://doi.org/10.1007/s40980-020-00061-8
7. Kim IK (2015) Population aging in Korea: social problems and solutions. J Sociol Soc Welf 26:107–123
8. Kim D-S (2005) Theoretical explanations of rapid fertility decline in Korea. Japanese J Popul 3:2–25
9. Lowe-Lee F (2009) Is Korea ready for the demographic revolution?
10. Phang H (2011) Issues and challenges facing population ageing in Korea: Productivity, economic growth, and old-age income security. J Comp Soc Welf 27:51–62. https://doi.org/10.1080/17486831.2011.532984
11. Mahmoudi KM (2017) Rapid decline of fertility rate in South Korea: causes and consequences. Open J Soc Sci 05:42–55. https://doi.org/10.4236/jss.2017.57004
12. South Korea's Ageing Democracy and its challenges.pdf
13. Shirota Y, Sari RF, Presekal A, Hashimoto T (2019) Visualization of time series data change by statistical shape analysis. In: 2019 16th Int Conf Qual Res QIR 2019—Int Symp Electr Comput Eng, 1–6. https://doi.org/10.1109/QIR.2019.8898291
14. Hasan MM, Sultana MI, Salma U, Lovely MLS (2020) Investigation of influential factors towards predicting birth rate in Bangladesh. Int Conf Emerg Trends Inf Technol Eng ic-ETITE 2020:1–6. https://doi.org/10.1109/ic-ETITE47903.2020.391
15. Shirota Y, Presekal A, Sari RF (2019) Visualization of time series data change on fertility rate and education in Indonesia provinces. In: 5th Int Conf Inf Manag ICIM 2019, pp 54–59. https://doi.org/10.1109/INFOMAN.2019.8714711
16. data.worldbank. https://data.worldbank.org

Automation in Agriculture: A Systematic Survey of Research Activities in Agriculture Decision Support Systems Using Machine Learning

Sushma Vispute and Madan Lal Saini

Abstract In this age of automation, Machine learning (ML) plays the main role in agriculture sector to suggest suitable advice, crop advice, which includes decisions of growing crops, and advice related to growing season for precision farming. This systematic literature review performs a review of 103 documents of different ML approaches to analyze the performance of algorithms and used features in the work of prediction of crop yield and decision support systems to solve agriculture problems. These 103 documents are retrieved from different electronic databases, for analysis. The paperwork presents methods, accuracy measures, and used agriculture parameters, to understand the existing work done by authors. According to analysis, most of the authors used N, P, and K values and type of soil, and most of the authors used classification techniques such as Support Vector Machine, Decision Trees, Regression techniques, Random Forest, and Naive Bayes algorithm; the most applied clustering algorithm in the existing work is K-means. As per the additional survey, the Convolution Neural Network (CNN) algorithm is used by most of the authors for image processing in their work. Also, survey shows that very few authors used associative classifiers and association rule mining techniques to solve the agriculture problems.

Keywords Associative classifier · Agriculture · Classification · Association rule mining · Clustering · Neural network · Decision support system · Machine learning · Convolution neural network

Madan Lal Saini: This author contributed equally to this work.

S. Vispute (✉) · M. L. Saini
Department of Computer Science and Engineering, Poornima University, Jaipur 303905, Rajasthan, India
e-mail: sushma.vispute@pccoepune.org

M. L. Saini
e-mail: madan.saini@poornima.edu.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*, Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_56

1 Introduction

Machine Learning techniques are used in many sectors, such as healthcare to predict the suitable treatment, supermarkets, manufacturing companies to analyze the customers' behavior like products used by customers. From several years Artificial Intelligence and ML techniques are also being applied in the agriculture-Farming sector to solve farmers' problems. Crop production is based on several parameters like seed type, climate, fertilizer used, weather and soil type, etc.

A problem with most of Indian farmers is, Lack of knowledge and Lack of proper assistance for precision Farming and so the objective of this Literature Review is to study and analyze existing Indian agriculture problems and solutions provided to these problems using Machine Learning techniques, to study and analyze different soil parameters which affect the agriculture production, to find the novel approach for proposed work.

The beauty of Machine Learning algorithm is to train the model using a training dataset and predict the class of new samples even though the new example is not completely matching with training samples. For example, where the training dataset contains CAT and DOG faces and predicted class for Tiger face as CAT.

There are three main categories of Machine Learning approaches. First, Supervised Learning includes learning from experience data, i.e., empirical data and its examples includes Classification, Regression (KNN, Decision Tree, and Linear Regression). Second Unsupervised Learning, i.e., Learning from observations given in the dataset, i.e., patterns in the dataset, its examples include, Clustering Techniques such as K-means, DBSCAN, third, Reinforcement Learning, i.e., learning from environment feedback in the form of penalty and rewards, for example, Deep Q Networks. Nowadays, deep learning algorithms are used for optimization of models because they attempt to learn by using a hierarchy of multiple layers [1].

2 Related Work

This work of literature review includes a survey of existing Indian agriculture problems and solutions provided to these problems using Machine Learning techniques, survey of different soil parameters which affect the agriculture production, and survey of different ML techniques to find the novel approach for proposed work.

Identified Research questions are:

Q1. Identify machine learning algorithm used for the Agriculture Support System.

Q2. Identify features used to design Agriculture Support System.

Q3. Identify model evaluation parameters and evaluation approaches used for the agriculture Support System.

Q4. Identify the Gaps in the field of Agriculture Support System.

2.1 Bibliography Analysis

Figure 1 shows bibliography analysis for distribution of papers used for the literature review.

Table 1 gives the count of documents referred for the survey on the basis of type of document and Table 2 gives the count of documents referred for the survey on the basis of publication year (Fig. 2).

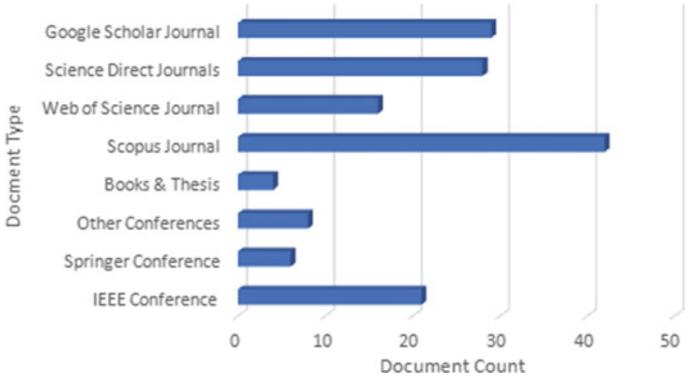


Fig. 1 Bibliography Analysis

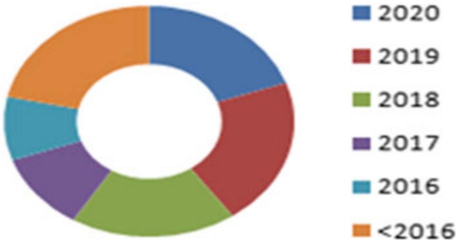
Table 1 Number of papers referred on the basis of type

Document type	# Documents
IEEE conference	21
Springer Conference	06
Web of science	16
Books and thesis	04
Science direct	28
Scopus	42
Google scholar	29
Others	08

Table 2 Distribution of documents based on the publication year

Publication year	#Documents
2020	20
2019	21
2018	19
2017	11
2016	09
<2016	22

Fig. 2 Number of documents referred on the basis of publication year



Sirsat et al. developed 20 different classification models for Classifying Indian agricultural soil parameters. They developed Soil nutrients N, P, K Classification model, Soil pH Classification model, model for Classification of Crop, model for Soil classification by type. These Classification problems are studied and implemented for the Marathwada dataset using Bagging, Boosting, Decision Tree (DT), K-Nearest Neighbor (KNN), Rule-Based (RB), Neural Network (NN), Random Forest (RF), and Support Vector Machine (SVM) models. Cohen kappa (k) in % is used by authors for measuring accuracy of these models. Model results are discussed below [2].

- Best k for Decision Tree Soil classifier using Weka tool is 97.82%,
- Best k for Random Forest Crop classifier using R language is 88.13%,
- Best k for Random Forest pH classifier using R language is 47.32%,
- Best k for Random Forest NPK classifier using R language is 33.6%,
- Best k for Random Forest OC-F classifier using Weka tool is 90.65%.

Sirsat et al. proposed 76 different models to predict soil fertility based on nutrients values of organic carbon (OC), phosphorus pentoxide (P_2O_5), Zn-Zinc, Fe-iron, and manganese (Mn) using different Regression Techniques such as Linear Regression (LR), Generalized Linear Regression (GLR), Least Square (LS), Partial Least Square (PLS), LASSO, Ridge, Neural Network, Deep Learning, SVM, Random Tree [3].

R2 accuracy measure is used by authors to find the best Regressor. Authors concluded following results of proposed models [3] (Table 3, Fig. 3).

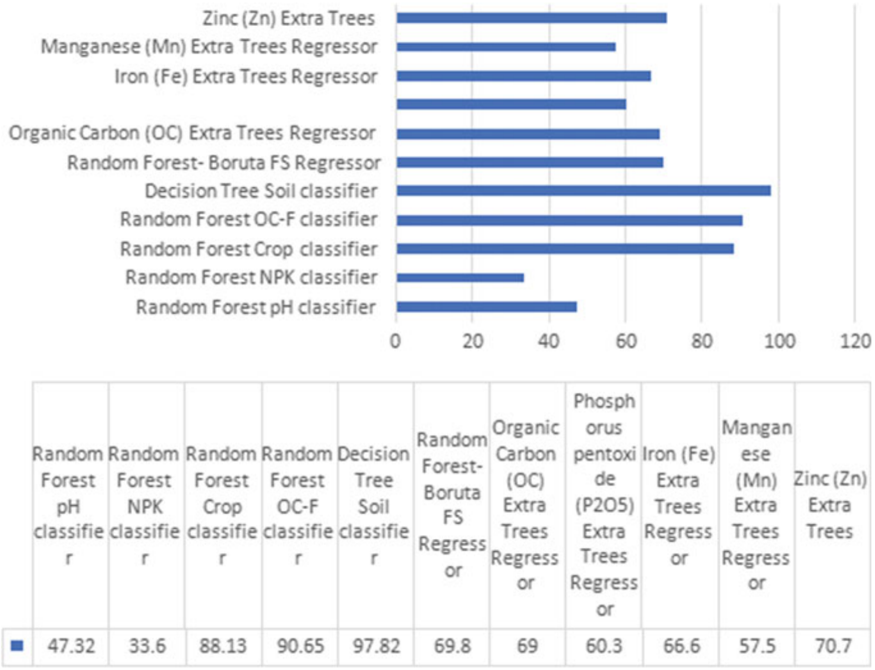
ZhaoyuZhai et al. represented a survey and challenges of agriculture (4.0) decision support systems (DSS). They did systematic survey of 13 representative DSS including their applications for planning missions, management of water resources, for controlling food waste, etc. [4].

Alexandre Barbosa et al. proposed a model for optimizing nutrient management for predicting crop yield response using CNN. Authors developed CNN with Early Fusion (EF), CNN with Late Fusion (LF), and 3D CNN and compared results with Multiple Linear Regression (MLR), Full Connected Network, Support Vector Network, and Random Forest models. Root Mean Square Error (RMSE) measure is used by authors to measure accuracy of CNN model. Results shows, CNN-LF with lowest error for nine tested fields (0.66), and CNN-RF with second best result (0.76) [5].

Suchithra et al. proposed a model for proper fertilizer utilization, to reduce the analysis time experts, and to improve quality of soil. In this work accuracy measures

Table 3 Comparative analysis of Indian agricultural soil parameters models accuracy

Model	Accuracy measure	Accuracy in %
Random forest pH classifier	Cohen Kappa	47.32
Random forest NPK classifier	Cohen Kappa	33.6
Random forest Crop classifier	Cohen Kappa	88.13
Random forest OC-F classifier	Cohen Kappa	90.65
Decision tree soil classifier	Cohen Kappa	97.82
Random forest—Boruta FS regressor	R^2	69.8
Organic carbon (OC) extra trees regressor	R^2	69
Phosphorus pentoxide (P_2O_5) extra trees regressor	R^2	60.3
Iron (Fe) extra trees regressor	R^2	66.6
Manganese (Mn) extra trees regressor	R^2	57.5
Zinc (Zn) extra trees	R^2	70.7



	Random Forest pH classifier	Random Forest NPK classifier	Random Forest Crop classifier	Random Forest OC-F classifier	Decision Tree Soil classifier	Random Forest-Boruta FS Regressor	Organic Carbon (OC) Extra Trees Regressor	Phosphorus pentoxide (P2O5) Extra Trees Regressor	Iron (Fe) Extra Trees Regressor	Manganese (Mn) Extra Trees Regressor	Zinc (Zn) Extra Trees
■	47.32	33.6	88.13	90.65	97.82	69.8	69	60.3	66.6	57.5	70.7

Fig. 3 Indian agricultural soil parameters models accuracy with data chart

used are Accuracy, Kappa, Precision, Recall, FScore, and results given by models are as follows [6].

- Soil Nutrient Classification for Gaussian radial basis function: 80% (Optimal neurons 50),
- pH classification for hyperbolic tangent function: 90% (Optimal neurons 150).

Himanshu Pant et al. proposed a model to enhance the precision of crop-fertility prediction using different supervised ML techniques. K-Means is used to identify quality and fertility of the Soil with levels 1, 2, and 3 for Nainital District dataset. Accuracy measures used for Classification problems are Precision, Recall, F1 Score, Support, Accuracy, and results are as below [7].

- SVM with 96.62% Accuracy (Best classifier among all),
- KNN with 91.01% Accuracy
- LR with 89.88% Accuracy
- LDA with 91.01% Accuracy.

Santhi et al. proposed a model to compare the categories of Farming and types of crops using crop and fertilizer recommendation methods based on soil test reports [8].

Manpriya et al. proposed a model for effective crop prediction technique for better crop production with more crop datasets. Deep NN with two hidden layers is used to predict appropriate crops for every district of India. 124 crops are included in the work. Performance parameters used by authors are Accuracy, Mean Absolute Error (MAE), and MSE. Sigmoid as activation function (SGD optimizer) is used for updating parameters and weights to reduce the loss function. Values of performance parameters are Accuracy with 99.19%, MAE with 0.0157, and MSE with 0.0078 [9].

Deshmukh et al. proposed a model for Soil Health Analysis and Soil quality prediction with N, P, and K Soil parameters. Results for soil quality prediction models and crop prediction models are shown in figure. CN2 Rule Inducer with accuracy of 0.94 declared as Best Classifier. Figure 4 shows performance comparison of Soil Quality and Crop Advice Prediction using different classifiers [10].

Labhade et al. developed a model to predict the outcomes based on the selected data and business requirements. Predictive Analytics is done using KNIME Tool and its results are as follows. Figure 5 shows Accuracy and Error rate for different classifiers using KNIME tool. As per analysis Logistic Regression method gives best accuracy for student datasets [11].

Viviliya et al. developed Hybrid model of J48 and Naive Bayes classifiers for recommending crops using ML techniques, to increase crop yield. Models are developed using dataset of parameters State, District, Crop year, Area, etc. and yield info from 1997 to 2015, Season, Temperature, Rainfall, Water requirement, and type of soil. J48 has given best accuracy 95.53% [12].

Devdatta et al. implemented a model of crop yield prediction using historical data by using machine learning on agriculture dataset and recommending fertilizers suitable for crop. Classification models using SVM and RF are built and authors used

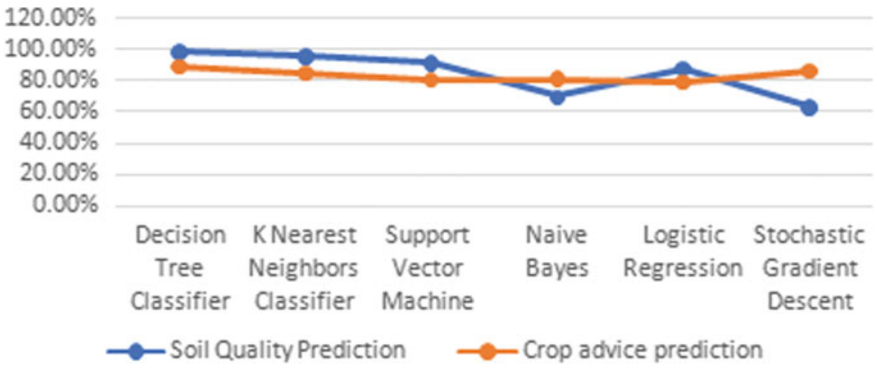


Fig. 4 Soil quality and crop advice prediction

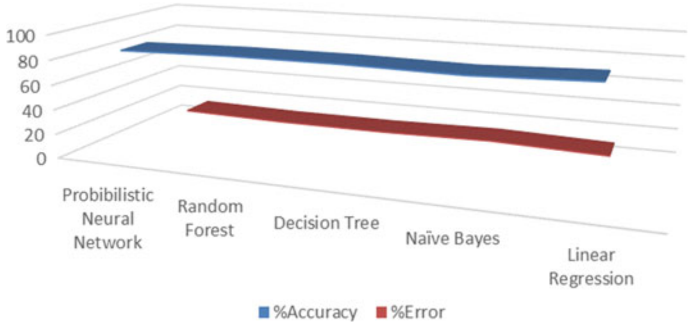


Fig. 5 Accuracy and error rate for different classifiers using KNIME tool

Precision, Recall, f1-score, and accuracy in % performance measuring parameters and discussed the results are as below [13].

- Soil Classification model using RF with accuracy of 86.35% and SVM with 73.75%.
- Crop Yield Prediction model using SVM with 99.47% accuracy and RF with 97.48% accuracy.

Rafael Hernández Moreno et al. presented a Multi-Layer Perceptron (MLP) ANN model with an input layer formed by soil parameters, an output layer with fertilizers and amendments. A GridSearchCV is used to test and optimize the model [14].

Archana et al. proposed a DSS model using Voting Based Ensemble Classifier. Voting based ensemble classifier for Crop recommendation (Random Forest Classifier, Naive Bayes Classifier, and CHAID Classifier) with input parameters, N, P, K, Temperature, and other soil parameter is built and got the 92% accuracy [15].

Rajak et al. developed a model for crop prediction using Ensemble technique (Majority Voting technique). In Ensemble technique different selected algorithms are SVM, Random Forest, NAÏVE Bayes, ANN- Multi-layer Perceptron [16].

Devotha et al. presented a review for survey of use of Characterization techniques in agriculture sector. They applied probabilistic and deterministic approaches, where the supervised algorithms are used in deterministic approaches, while the unsupervised algorithms are used in probabilistic approaches [17].

Srivastava et al. presented survey paper to electorate on different Clustering Techniques such as DBSCAN, Agglomerative, K-means, EM algorithms for Agriculture applications to bring a good advancement in the agricultural area for Forecasting Pollution, Combined Classification of Soil with GPS [18].

Bouighoulouden et al. proposed a model using PCA for reduction of the features and K-means implemented on Rstudio, Orange DM tools to identify groups of productive and non-productive yield [19].

Dr. Madhavi Gudavalli et al. applied Clustering on Wheat seed dataset using different clustering techniques. 3 clusters are formed Kama, Rosa, Canadian with pair of attributes using R tool, authors reported that k-mean is good for large datasets and Hierarchical is good for small datasets [20]

Priya et al. built a model for depiction of management zones and soil dataset analysis using K-means, GK clustering, and Farthest First (obtained Best-faster) Algorithms [21].

Utkarsha et al. developed Modified K-Means Algorithm and used it for crop prediction. District, zone, and selection of seasons, max temperature, min temperature, soil type, and average rainfall are considered for training the model. Work shows comparison of k-Means++ and k-Means with modified k-Means on Crop data. Modified k-Means gave the maximum quality clusters, maximum accuracy count, and correct prediction of crop [22].

Silas et al. used Association Rule Mining and Clustering Techniques for Tea Production prediction in Kenya country. Dataset contains 156 tea production records from year 2003 to 2015. Clustering techniques are used to form the groups of similar productions using (SPSS) K-Means [23].

Majumdar et al. presented analysis using different ML techniques such as Multiple LR, CLARA, PAM, and Modified DBSCAN to identify optimal parameters to maximize crop production. Modified DBSCAN was declared as a Best to cluster the data having similar rainfall, temperature, and soil type [24].

Vandana et al. proposed model for crop production and US arrest dataset analysis. Techniques used are Hybrid K- means which declared as a Best. Elbow, Gap Statistic, Silhouette Methods are used to select optimal "K" value [25].

Aurelia-Vasilicalana et al. used clustering methods for Organic farming patterns analysis. Work identified three possible clusters using clustering methods [26].

Chunjiang et al. built a model using Frequent Pattern Tree for mining association rules with multiple inputs of minimum supports (MSDMFIA). It overcomes the problem of single minimum support used in tradition method [27].

Geetha et al. used Apriori algorithm to assess different association algorithms and used them into a soil science database to identify meaningful relationships [28].

Kane et al. proposed model for Classification of home loan sales in an Irish retail banking using Association Rules. Associative classifier models used are CMAR, Classification Based Association (CBA), and SPARCCC [29].

Vasoya et al. proposed distributed model based on distributed and parallel computing for large dataset association rule mining to find frequent patterns in less time. Clustering process is used to divide large data into number of clusters and these clustered data are used for mining process [30].

Thakkar et al. used Association rule mining algorithms like Apriori and classification techniques like ID3 and C4.5, to solve agriculture crops problems [31]. Khan and Singh [32], presented survey of Association Rule Mining methods for agriculture problems. Survey represents techniques used to solve problems using the Partition Algorithm, Apriori, Pincer search Algorithm, FP-Tree Growth Algorithm, Dynamic Itemset Counting Algorithm [32].

Mishra et al. presented survey of Associative Classifiers (CBA, CMAR, MCAR, and GARC) used on Soil dataset of Bhopal M.P District [33]. Sun et al. [34] presented an Overview of Associative Classifiers. The conventional classification system such as C4.5 is compared with associative classifier. Total 27 UCI datasets are used for comparison [34].

Prachitee et al. proposed a model of Classification Technique using Associative Classifier based on the Neural Network system (NNAC) to improve its accuracy. NNAC system performance is compared with the Classification Based Association on four different datasets from UCI repository [35].

Soni et al. proposed solution for Health Care domain using Associative Classifiers to predict the disease with some suitable treatments. Authors used class rule mining—Associative Classification (AC), classification Association rule (CAR) techniques [36]. Classifier to assist the physician to find association among patient parameters (e.g., personal data, medical tests,) have also been developed, and advanced association rule mining with classifiers are used to develop models of an AC based on positive and negative rules, Temporal AC, AC using Fuzzy Association Rule, Weighted AC [36].

Jinubala et al. proposed a model to classify Pest Level based on whether data using Constraint-based AC (Accuracy 92%) and Traditional method accuracy of 59% [37]. Mattieva and Kavšeka [38], proposed a model using associative classification techniques such as AC based on strong association rules. Average accuracy given by model is 91.3%. Experiments are done on 15 UCI ML D/B repositories [38].

Li Yu Hu et al. work presented Novel CBA-based method: MMSCBA, (multiple minimum supports (MMSs)) [39]. Dalvi et al. [40], proposed a Ontology-based model for agricultural (IR) using NLP, to extract knowledge in Marathi language [40].

Pai et al. presented ML models for Identification of Kannada Farmer’s Query using a speech recognition system for agricultural dataset in Kannada language. The dataset consists of the name of the crops and name of the districts of Karnataka state. MFCC is the most prominent feature extraction method used in speech recognition. MFCC for CROP, District Data [41].

Savant et al. presented survey of existing system of Maharashtra Government, Survey of clustering Techniques, and Classification of farmer’s feedback [42]. Vispute et al. [43] proposed a model for automatic personalized Marathi content generation in Marathi language using LINGO algorithm. Work has experimented on five different datasets and personalization is done using “Time Session”, “Number of hits” and Bookmark methods [43].

Vispute et al. extended previous work using HADOOP parallel system platform for Marathi dataset [44]. Vispute et al. [44], developed a model for categorizing Marathi text documents automatically for dataset of three categories- Health Programs, Tourism, and Maharashtra festivals using Lingo Clustering algorithm. Dataset contains 107 Marathi documents [45].

Sonigara et al. built a model for effective information retrieval system to input the data in heterogeneous forms and represent it into a common format, i.e., a text file, and categorizing Marathi data automatically using LINGO algorithm [46].

Tayal and Meena developed parallel system solution using the MapReduce approach on HADOOP platform for associative classification and experimented on six datasets available on UCI repositories. To provide solution to problems they developed two algorithms MRMCAR-F and MRMCAR-L [47]. Figure 6 shows accuracy comparison of proposed association classification techniques by Devendra et al. for six different datasets of UCI data repository.

Figure 7 shows comparison of time required to execute different associative classification techniques proposed by Tayal and Meena [47].

Dang Nguyen et al. proposed an efficient constraint-based CARs model with the item set constraint. To test the performance of novel model authors used 14 different datasets like Adult, Breast, German, Chess, Connect4, etc. Figure 8 shows proposed models for adult dataset [48].

Wang et al. proposed an improved model using dynamic property in the associative classification [49]. Villuendas-Rey et al. [50] used and evaluated the Naïve Associative Classifier on financial dataset for simple, transparent, and accurate classification [50].

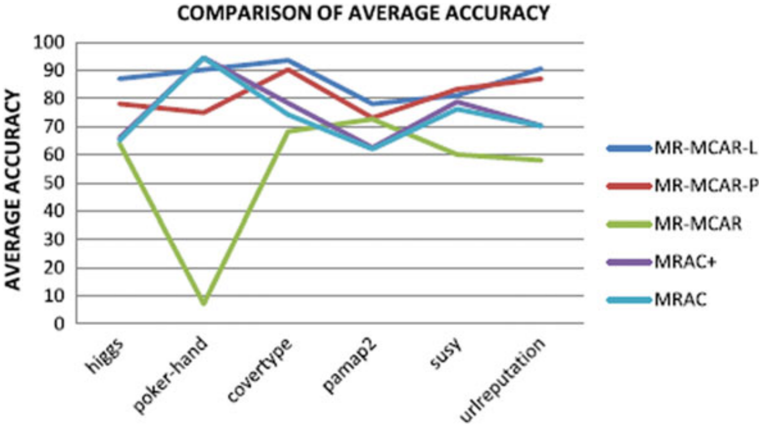


Fig. 6 Comparison of proposed association classification techniques for six different datasets of UCI data repository [47]

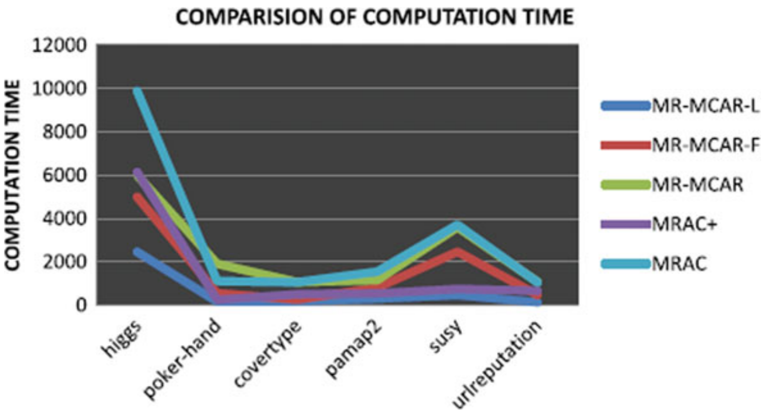
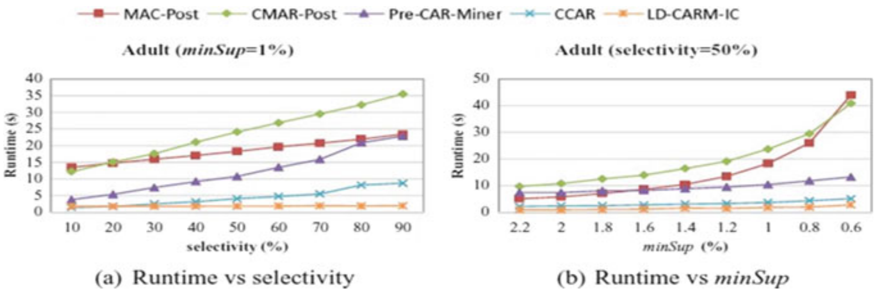


Fig. 7 Comparison of execution time with the proposed associative classification techniques [47]



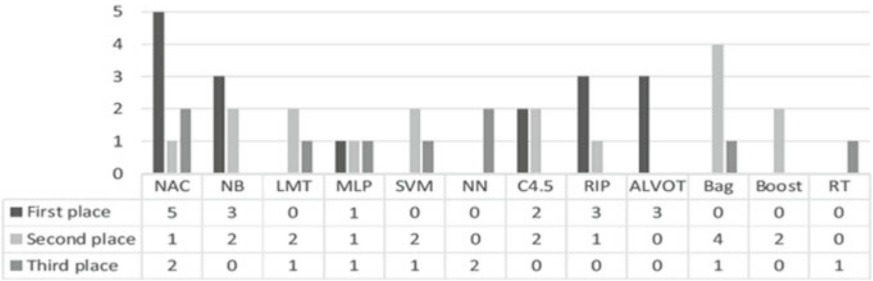
(a) Runtime vs selectivity

(b) Runtime vs *minSup*

Fig. 8 Comparison of execution time required for different proposed models for adult dataset [48]

Figure 9 shows the overall AUC results of NAC, compared with other classifiers. It shows that NAC outperforms as compared to other algorithms.

Chen et al. proposed an efficient classification approach, Principal Association Mining to design a compact classifier for generating reduced association rules [51].



	NAC	NB	LMT	MLP	SVM	NN	C4.5	RIP	ALVOT	Bag	Boost	RT
■ First place	5	3	0	1	0	0	2	3	3	0	0	0
■ Second place	1	2	2	1	2	0	2	1	0	4	2	0
■ Third place	2	0	1	1	1	2	0	0	0	1	0	1

Fig. 9 AUC results of NAC, compared with other classifiers [50]

Padillo et al. [52] introduced a new Library of JAVA language for Associative classification, i.e., LAC. This library package includes the full taxonomy of associative classification paradigm [52]. Loan et al. [53], developed a new model for extracting class-association rules [53]. Antonell et al. [54] used fuzzy-frequent pattern mining algorithm to proposed a novel classification model. Authors tested the new approach on 17 datasets and represented comparative analysis. New model gave better results than existing [54]. Hadi et al. [55] proposed efficient model for text classification which combines features of Naïve Bayes and associative classifiers [55]. Thasleena et al. [56] developed an efficient classifier for XML documents using associative classifier to overcome the drawback of the existing technology [56]. Mattieva et al. [57], proposed simple classification with "strong" class-association rules to improve the classifier performance with good accuracy [57].

Nguyena et al. and Wang et al. proposed hybrid and an efficient method to solve problem using associative classification techniques [58, 59]. Villuendas-Reya et al. proposed new model NAC, based on Associative classifier, and tested and evaluated model on financial dataset [60].

In the next literature survey of agricultural decision support systems for precision farming we compared different ML and Deep Learning algorithms and explored possible uses of these algorithms to solve multiple problems related to farming.

Many algorithms like SVM, Random Forest, and CNN were used to detect plant diseases. The result shows that CNN detects a greater number of diseases of plants with high accuracy [61–65].

In scenarios where there is huge difference between size or color difference between crop and weed, image processing-based algorithm works well. Survey tells that CNN performs better than the SVM and ANN due of its ability of learning in depth to learn related features from the image dataset. ANN is very accurate but requires huge amounts of training data and is slower [66–70].

For weather forecasting research shows that different models such as ANN, CNN, and Recurrent NN can be used. Out of these models, Long Short-Term Memory LSTM (type of RNN) works exceptionally well for sequential data of weather prediction [71–75].

Many algorithms like ARIMA, SARIMA, and RNN algorithms such as LSTM and Gated Recurrent Units (GRU) can be used to predict agricultural prices. The results show that in general LSTM models perform better than others with higher data while ARIMA and SARIMA can perform reasonably well even with less data [76–80].

The next survey of work shows, solution to a variety of problems like prediction of soil fertility level, disease detection, prediction of yield based on weather conditions, identifying correct action during farming in different situations, etc. [81–85].

3 Common Findings from Literature Review

3.1 Results and Common Methodology

Most commonly used methodology by authors is given in the below Fig. 10. It includes following basic steps to develop a model for solving problems.

Table 4 shows most used machine learning algorithms in the existing work with efficient model details. (Answer of question1).

In the review of existing work, it is found that Agriculture decision support systems are developed to provide decisions about single areas of farming such as recommendations for Crop yield prediction, recommendation for fertilizer, etc. by using different machine learning techniques mentioned in Table 4.

In the survey, it is found that very little work is done to provide solutions to the agriculture problems using associative classifiers, only three paperwork shows solutions to agriculture problems using associative classifiers. This existing work only provides a single decision to farmers at a time by considering different agriculture parameters such as N, P, K, Ph, Crop year, Rainfall, etc. So, the more effective agriculture decision support system needs for precision farming.

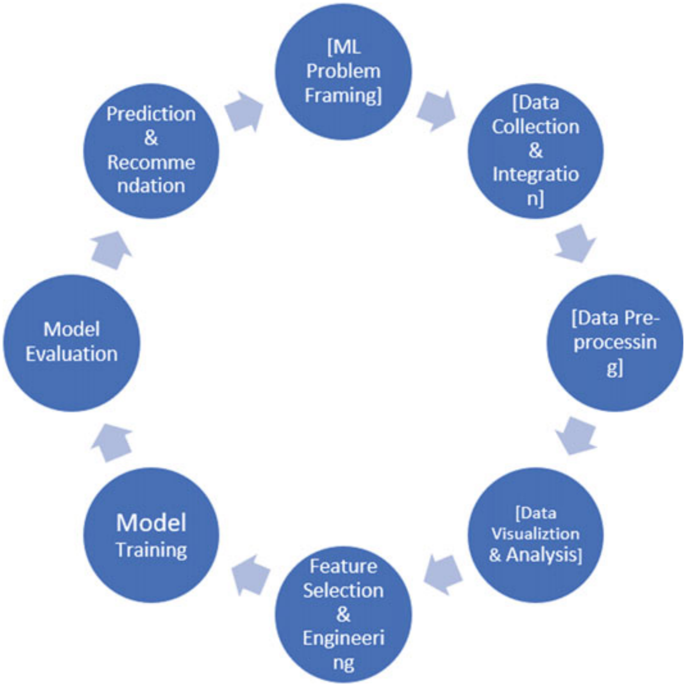

[ML Problem Framing]
[Data Collection & Integration]
[Data Pre-processing]
[Data Visualiztion & Analysis]
Feature Selection & Engineering
Model Training
Model Evaluation
Prediction & Recommendation

Fig. 10 Common flow diagram for Developing ML Model

Table 4 Algorithms used by most of the existing work with best model details

Algorithms	Frequency count	Efficient model details
Classification: RT, RF, NB, DT, SVM	29	Random forest is better in most of the work
Neural network: CNN, DNN	27	CNN with Relu and optimized parameters
Clustering: K-means	07	K-Means by most of the work for efficient analysis
Clustering: agglomerative, DBSCAN	02	
ARM: Apriori	04	Apriori for appropriate results
ARM: FP-growth	01	
Associative classifier	17 (only 2 for Agri problems)	Hybrid associative classifier

Features used in most of the work (Answer to question 2) are Sulfur, Magnesium, potassium, zinc, nitrogen, calcium, boron, and Phosphorus, pH-value State, District, Crop year, Season, Area, Production and yield details, Rainfall details, Temperature details, Groundwater level, Water availability, type of soil, Organic carbon (OC).

Most used evaluation parameters (Answer of question 3) are Accuracy (36 times), Kappa (8), Precision (27), Recall (27), FScore (24), RMSE(6), R^2(5), WCSS (9), Support and confidence(5).

4 Conclusion

This systematic literature review showed that the work in the referred documents those used a several features, depending on the research type and requirements with the selected dataset. Most of the work is done for prediction of yield and applied machine learning algorithms but on different features. Also, work is done for plant disease prediction, weed detection. Selected features are dependent on the objectives of the research. The best model can be identified by testing models with more features and fewer features and also models with different ML techniques. According to survey study and analysis, most of the authors used rainfall, temperature, and type of soil, and most preferred classification algorithms are Neural Networks, Regression techniques, SVM, Random Forest, andRandom Forest worked better in most of the work. The most applied clustering algorithm is K-means for finding efficient solution to problem.

As per the additional survey, Convolution Neural Networks (CNN) with optimized parameters is used by most of the authors for image processing and classification and then another widely used DL algorithm is Deep Neural Networks (DNN). Also, survey shows that very few authors (only 2) used associative classifiers and association rule mining techniques to solve the agriculture problems.

References

1. Han J, Kamber M (2011) Data mining: concepts and techniques. Morgan Kaufmann, 3rd edn. A volume in The Morgan Kaufmann Series in Data Management Systems
2. Sirsat MS, Cernadas E, Fern´andez-Delgado M, Khan R (2017) Classification of agricultural soil parameters in India. Comp Elect Agri 135:269–279
3. Sirsat EC, Fern´andez-Delgado S, Barro S (2018) Automatic prediction of village-wise soil fertility for several nutrients in India using a wide range of regression methods. Comp Elect Agri 154:120–133
4. ZhaoyuZhai J, Martínez J-F, Beltran V, Martínez NL (2020) Decision support systems for agriculture 4.0: Survey and challenges. Comp Elect Agri Science Direct 170:105256
5. Barbosa A, Trevisan R, Hovakimyan N, Martin NF (2020) Modeling yield response to crop management using convolutional neural networks. Comp Elect Agri Sci Direct 170:105197
6. Suchithra MS, Pai ML (2019) Improving the prediction accuracy of soil nutrient classification by optimizing extreme learning machine parameters. Sci Direct, Info Process Agri 7(1):72–82
7. Pant H, Lohani MC, Bhatt A, Pant J, Joshi A (2020) Soil quality analysis and fertility assessment to improve the prediction accuracy using machine learning approach. Int J Adv Sci Tech 29(3):10032–10043
8. Santhi P, Priyanka T (2020) Smart India agricultural Info retrieval system. Int J Adv. Sci Tech 29(7):1169–1175
9. Manpriya D, Jindal V (2020) Crop prediction using deep neural network. Int J Mech Product Eng Res Develop 3:2249–6890
10. Deshmukh S, Dhannawat D, Dalvi M, Gawali P, Vispute SR, Kekane S (2019) Application of data analytics in agriculture sector for soil health analysis: Literature review. In: 5th International Conference on Computing, Communication, Control and Automation (ICCUBEA-2019), pp 1–4. https://doi.org/10.1109/ICCUBEA47591.2019.9129104
11. Labhade D, Lakare N, Mohite A, Bhavsar S, Vispute S, Mahajan G (2019) An overview of machine learning techniques and tools for predictive analytics. Asian J Conv Tech 5(3):63–66
12. Viviliya B, Vaidhehi V (2019) The design of hybrid crop recommendation system using machine learning algorithms. Int J Innov Tech Exploring Eng 9(2):4305–4311
13. Devdatta AB, Mahagaonkar S (2019) Prediction of crop yield and fertilizer recommendation using machine learning algorithms. Int J Eng Appl Sci Tech 4(5):371–376
14. Hernández Moreno R, Garcia O, Luis Alejandro Arias R (2018) Model of neural networks for fertilizer recommendation and amendments in pasture crops. In: 2018 IEEE, 978-1-5386-9459-6/18/$31.00.
15. Archana K, Saranya KG (2020) Crop yield prediction, forecasting and fertilizer recommendation using voting based ensemble classifier. Int J Comp Sci Eng 7(5):1–4
16. Rajak RK, Pawar A, Pendke M, Shinde P, Rathod S, Devare A (2017) Crop recommendation system to maximize crop yield using machine learning technique. Int Res J Eng Tech 4(12):950–953
17. Devotha G. Nyambo ETL, Yonah ZQ (2019) A review of characterization approaches for smallholder farmers: towards predictive farm typologies. Hindawi: Scient World J Wiley, 1–10
18. Srivastava V, Aggarwal KK, Srivastava AK (2019) A revisit to clustering techniques with its application in agriculture sector. HEB 3(1):1–1
19. Bouighoulouden A, Kissani I (2020) Crop yield prediction using K-means clustering, school of science and engineering. Al Akhawayn University, Spring
20. Gudavalli M, Vidyasree P, Viswanadha Raju S (2017) Clustering analysis for appropriate crop prediction using hierarchical, fuzzy C-means, k-means and model based techniques. Int J Adv Eng Res Develop 4(11):1233–1242
21. Krishna Priya CB, Venkateswari S (2018) Delineation of management zones in precision agriculture using different clustering algorithms. Int J Appl Eng Res 13(22):15951–15955
22. Utkarsha PN, Adhiya KP (2016) Evaluation of modified k-means clustering algorithm in crop prediction. Int J Adv Comp Res 4(16):799–807

23. Silas NM, Nderu L (2017) Prediction of tea production in Kenya using clustering and association rule mining techniques. American J Comp Sci Info Tech 5:1–7
24. Majumdar J, Naraseeyappa S, Ankalaki S (2017) Analysis of agriculture data using data mining techniques: application of big data. J Big Data 4(20):1–15
25. Vandana B, Sathish Kumar K (2019) Hybrid k mean clustering algorithm for crop production analysis in agriculture. Int J Inno Techn Explo Eng 9(2S):9–12
26. Aurelia-Vasilicalana ET, Dobrea C, Soarea E (2015) Organic farming patterns analysis based on clustering methods, Science Direct. Agri Agricultural Sci Procedia 6:639–646
27. Zhao Chunjiang W, Huarui SX, Baozhu Y (2010) Algorithm for mining association rules with multiple minimum supports based on FP-Tree. N Z J Agric Res 50(5):1375–1381. https://doi.org/10.1080/00288230709510425
28. Geetha MCS (2015) Implementation of association rule Mining for different soil types in agriculture. Int J Adv Res Comp Comm Eng 4(4):520–523
29. Kane C (2018) Classification using association rules. Technological University Dublin, ARROW@TU Dublin
30. Vasoya A, Koli N (2016) Mining of association rules on large database using distributed and parallel computing. In: 7th International Conference on Communication, Computing and Virtualization.Proccedia Computer Science, 79, 221–230
31. Rahul GT, Kayasth M, Desai H (2014) Rule based and association rule mining on agriculture dataset. IJRICCE
32. Khan F, Singh D (2014) Association rule mining in the field of agriculture: a survey. Int J Inno Res Comp Comm Eng 2(11):6381–6384
33. Mishra AK, Sharma P (2014) A review on associative classification data mining approach in agricultural soil land. Int J Modern Trends in Eng Res 1(4):65–69
34. Sun Y, Andrew KC, Wong F, Wang Y, Member IEEE (2006) An overview of associative classifiers. In: International Conference on Data Mining, DMIN, Las Vegas, Nevada, USA, 1–7.
35. Prachitee B. Sheetal S, Dhande S (2011) A classification technique using associative classification. Int J Comp Appl 20(5):20–28
36. Soni S, Vyas OP (2010) Using Associative Classifiers for Predictive Analysis in Health Care Data Mining. International Journal of Computer Applications, 2010, 4(5), 33–37.
37. Jinubala V, Raj L (2018) Mining pest level based on weather using associate classification. Pestology Wiley XLII(3):1–8
38. Mattieva J, Kavšeka B (2020) A compact and understandable associative classifier based on overall coverage. Procedia Computer Science170:1161–1167
39. Hu L-Y, Hu Y-H, Tsai C-F, Wang J-S, Huang M-W (2016) Building an associative classifier with multiple minimum supports. Springer Plus 5(528):1–19
40. Dalvi P, Mandave V, Gothkhindi M, Patil A, Kadam S, Pawar S (2016) Ontology extraction for agriculture domain in Marathi language using NLP techniques, ICTACT-J Soft Comp 7(1):1359–1365
41. Pai A, Hegde S (2019) Study on machine learning for identification of farmer's query in Kannada language. Int J Comp Appl (0975–8887) 178:40–47
42. Savant V, Shinde A, Yedle B, Pantawane S, Vispute SR, Pede SV (2015) A survey on farmer's need and feedback, IJIRCCE
43. Vispute SR, Kanthekar S, Kadam A, Kunte C, Kadam P (2014) Automatic personalized Marathi content generation. In: International Conference on Circuits, Systems, Communication and Information Technology Applications (CSCITA), pp 294–299. https://doi.org/10.1109/CSCITA.2014.6839275
44. Vispute SR, Patil S, Sangale S, Padwal A, Ukarde A (2015) Parallel processing system for Marathi content generation. International Conference on Computing Communication Control and Automation 2015:575–579. https://doi.org/10.1109/ICCUBEA.2015.118
45. Vispute SR, Potey MA (2015) Automatic text categorization of marathi documents using clustering technique. In: 15th International Conference on Advanced Computing Technologies (ICACT), pp 1–5, https://doi.org/10.1109/ICACT.2013.6710543

46. Prachi S, Kirti P, Pooja N, Alisha S, Sushma V (2018) Automatic integration and clustering of Marathi documents in different formats for effective information retrieval. In: Proceedings of International Conference on Recent Advancement on Computer and Communication, Lecture Notes in Networks and Systems. https://doi.org/10.1007/978-981-10-8198-9_36
47. Tayal DK, Meena K (2020) A new MapReduce solution for associative classification to handle scalability and skewness in vertical data structure. Future Generation Comp Syst 103:44–57
48. Nguyen D, Loan TT, Nguyen BV, Pedry W (2016) Efficient mining of class association rules with the itemset constraint. J Know Based Syst 103:73–88
49. Wang X, Yue K, JiaNiu W, Shi Z (2011) An approach for adaptive associative classification. Expert Syst Appl ScienceDirect 38:11873–11883
50. YennyVilluendas-Rey CF, Rey-Benguría Á-S, Camacho-Nieto O, Yáñez-Márquez C (2017) The Naïve Associative Classifier (NAC): a novel, simple, transparent, and accurate classification model evaluated on financial data. Neurocomputing J 265:105–115
51. Chen F, Wang Y, Li M, Harris W, Tian J (2014) Principal association mining: an efficient classification approach. Knowledge-Based Syst J 67:6–25
52. Padillo F, Luna JM, Ventura S (2019) LAC: library for associative classification. Knowledge-Based Systems J 193:105432
53. Loan TT, Nguyen B, Vo B, Hong T-P, Thanh HC (2012) Classification based on association rules: A lattice-based approach. J Expert Syst Appl 39:11357–11366
54. Antonelli M, Ducange P, Marcelloni F, Segatori A (2015) A novel associative classification model based on a fuzzy frequent pattern mining algorithm. J Expert Syst Appl 42:2086–2097
55. Hadi W, Qasem A, Al-Radaideh SA (2018) Integrating associative rule-based classification with Naïve Bayes for text classification. J Appl Soft Comp 69:344–356
56. Thasleena NT, Varghese SC (2014) Enhanced associative classification of XML documents supported by semantic concepts. Int Conf Information Comm Tech 46:194–201
57. Mattiev J, Kavšeka B (2020) A compact and understandable associative classifier based on overall coverage. In: International Workshop on Statistical Methods and Artificial Intelligence (IWSMAI), April 6–9, Warsaw, Poland, 170, 1161–1167
58. Dang Nguyena, Loan T.T. Nguyenb, Bay Vo, Witold Pedryczf. Efficient mining of class association rules with the itemset constraint, Knowledge- Based Systems journal, 2016, doi:https://doi.org/10.1016/j.knosys.2016.03.025
59. Wang X, Yue K, JiaNiu W, Shi Z (2011) An approach for adaptive associative classification. Expert Syst Appl ScienceDirect J. https://doi.org/10.1016/j.knosys.2016.03.025.
60. YennyVilluendas-Reya CF, Rey-Benguría Á, Santiagoc O-N, Yáñez-Márquez C (2017) The Naïve Associative Classifier (NAC): a novel, simple, transparent, and accurate classification model evaluated on financial data. Neurocomputing 000:1–11
61. Sharma P, Berwal YPS, Ghai W (2018) KrishiMitr (farmer's friend): using machine learning to identify diseases in plants. In: IEEE International Conference on Internet of Things and Intelligence System (IOTAIS), pp 29–34. https://doi.org/10.1109/IOTAIS.2018.8600898
62. Ramesh S, et al (2018) Plant disease detection using machine learning. In: International Conference on Design Innovations for 3Cs Compute Communicate Control (ICDI3C), pp. 41–45. https://doi.org/10.1109/ICDI3C.2018.00017
63. Ferentinos K (2018) Deep learning models for plant disease detection and diagnosis. Comput Electron Agric 145:311–318
64. Gaikwad VP, Musande V (2017) Wheat disease detection using image processing. In: 1st International Conference on Intelligent Systems and Information Management (ICISIM), pp 110–112. https://doi.org/10.1109/ICISIM.2017.8122158
65. Barure S, Mahadik B, Thorat M, Kalal A (2020) Disease detection in plant using machine learning. IRJET 7(3), Mar
66. Irías Tejeda J, Castro R (2019) Algorithm of weed detection in crops by computational vision. In: 2019 International Conference on Electronics, Communications and Computers (CONIELECOMP), Cholula, Mexico, pp 124–128. https://doi.org/10.1109/CONIELECOMP.2019.8673182

67. Kumaraswamy R, et al (2019) Performance comparison of weed detection algorithms. In: 2019 International Conference on Communication and Signal Processing (ICCSP), Chennai, India, pp 0843–0847. https://doi.org/10.1109/ICCSP.2019.8698094
68. Umamaheswari S, Arjun R, Meganathan D (2018) Weed detection in farm crops using parallel image processing. In: IEEE 2018 Conference on Information and Communication Technology (CICT), Jabalpur, India, pp 1–4. https://doi.org/10.1109/INFOCOMTECH.2018.8722369
69. Hameed S, Amin I (2018) Detection of weed and wheat using image processing. In: 5th International Conference on Engineering Technologies and Applied Sciences (ICETAS), Bangkok, Thailand, pp 1–5. https://doi.org/10.1109/ICETAS.2018.8629137
70. Barrero O, Rojas D, Gonzalez C, Perdomo S (2016) Weed detection in rice fields using aerial images and neural networks. In: 2016 XXI Symposium on Signal Processing, Images and Artificial Vision (STSIVA), Bucaramanga, pp 1-4. https://doi.org/10.1109/STSIVA.2016.7743317
71. Mishra M, Srivastava M (2014) A view of artificial neural network. In; International Conference on Advances in Engineering & Technology Research (ICAETR—2014), pp.1–3. https://doi.org/10.1109/ICAETR.2014.7012785
72. Albawi S, Mohammed TA, Al-Zawi S (2017) Understanding of a convolutional neural network. In: International Conference on Engineering and Technology (ICET), pp 1–6. https://doi.org/10.1109/ICEngTechnol.2017.8308186
73. Denny Prabowo Y, Warnars HLHS, Budiharto W, Kistijantoro AI, Heryadi Y, Lukas (2018) LSTM and simple RNN comparison in the problem of sequence to sequence on conversation data using Bahasa Indonesia. In: Indonesian Association for Pattern Recognition International Conference (INAPR-2018), pp 51–56. https://doi.org/10.1109/INAPR.2018.8627029
74. Salman AG, Kanigoro B, Heryadi Y (2015) Weather forecasting using deep learning techniques. In: International Conference on Advanced Computer Science and Information Systems (ICACSIS), pp 281–285. https://doi.org/10.1109/ICACSIS.2015.7415154
75. Naveen L, Mohan HS (2019) Analyzing impact of weather forecasting through deep learning in agricultural crop model predictions. Int J Appl Eng Res 14:4379–4386
76. Selvanayagam T, Suganya S, Palendrarajah P (2019) Agro-genius: crop prediction using machine learning. Int J Inno Sci Res Tech 4(10):243–249
77. Sabu KM, Manoj Kumar TK (2020) Predictive analytics in Agriculture: Forecasting prices of Arecanuts in Kerala. Procedia Comp Sci 171:699–708
78. Peng Y, Hsu C, Huang P (2015) Developing crop price forecasting service using open data from Taiwan markets. In: Conference on Technologies and Applications of Artificial Intelligence (TAAI), Tainan, Taiwan, pp 172–175. https://doi.org/10.1109/TAAI.2015.7407108. ©2015 IEEE
79. Vohra NP, Khatri SK (2019) Decision making support system for prediction of prices in agricultural commodity. In: Amity International Conference on Artificial Intelligence (AICAI), Dubai, United Arab Emirates, pp 345–348. https://doi.org/10.1109/AICAI.2019.8701273 ©2019 IEEE
80. Kurumatani K (2018) Time series prediction of agricultural products price based on time alignment of recurrent neural networks. In: 17th IEEE International Conference on Machine Learning and Applications (ICMLA), Orlando, FL, USA, pp 81–88. https://doi.org/10.1109/ICMLA.2018.00020
81. Mhudchuay T, Kasetkasem T, Attavanich W, Kumazawa I, Chanwimaluang T (2019). Rice cultivation planning using a deep learning neural network. In: 16th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), pp 822–825. https://doi.org/10.1109/ECTI-CON47248.2019.8955227
82. Rajesh R, Saradhambal D, Latha S (2018) Plant disease detection and its solution using image classification. Int J Pure Appl Math 119:879–884
83. Mishra DK, Veenadhari S, Singh CD (2011) Soybean productivity modelling using decision tree algorithms. Int J Comp Appl

84. Sehgal A, Mathur S (2019) Plant disease classification using soft computing supervised machine learning. In: 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA), pp 75–80. https://doi.org/10.1109/ICECA.2019.8822213
85. Mehta P, Shah H, Kori V, Vikani S, Shukla, Shenoy M (2015) Survey of unsupervised machine learning algorithms on precision agricultural data. In; International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS-2015). pp 1–8. https://doi.org/10.1109/ICIIECS.2015.7193070
86. van Klompenburga T, Kassahuna A, Catal C (2020) Crop yield prediction using machine learning: A systematic literature review. Comp Elect Agri, Sci Direct 177:105709
87. Bacco M, Barsocchi P, Ferro E, Gotta A, Ruggeri M (2019) The digitisation of agriculture: a survey of research activities on smart farming. Array 3–4:100009
88. Jha K, Doshi A, Patel P, Shah M (2019) A comprehensive review on automation in agriculture using artificial intelligence. Artificial Intell Agri Sci Dir 2:1–12
89. Yuzhen L, Young S (2020) A survey of public datasets for computer vision tasks in precision agriculture. Comput Electron Agric 178:105760
90. Padarian J, Minasny B, McBratney AB (2019) Machine learning and soil sciences: a review aided by machine learning tools. Soil, EGU, 6(1):35–52
91. Bachhav NB (2012) Information needs of the rural farmers: a study from Maharashtra, India: a survey. Digital Commons @University of Nebraska. Library Phil Pract (e-journal) 866:1–13
92. Young A, Mahan J, Dodge W, Payton P (2020) BLOB-based AOMs: a method for the extraction of crop data from aerial images of cotton. MDPI-Agriculture. https://doi.org/10.3390/agriculture10010019
93. Jain L et al (2017) Cloud-based system for supervised classification of plant diseases using convolutional neural networks. In: IEEE International Conference on Cloud Computing in Emerging Markets (CCEM), pp 63–68. https://doi.org/10.1109/CCEM.2017.22
94. Kokane P, Vispute S, Aarti Kalekar KB, Kamble M (2018) Automated generation, calculation of the village soil fertility index and analysis of soil health card. IJNTSE-ISSN5
95. Medar R, Rajpurohit VS, Shweta S (2019) Crop yield prediction using machine learning techniques. In: IEEE 5th International Conference for Convergence in Technology (I2CT-2019), pp. 1–5. https://doi.org/10.1109/I2CT45611.2019.9033611
96. Deshmukh PR, Badnuke MR (2012) Infected leaf analysis and comparison by Otsu threshold and k-means clustering. Int J Adv Res Comp Sci Soft Eng 2(3):449–452
97. Sun L, Yang Y, Hu J, Porter D, Marek T, Hillyer C (2017) Reinforcement learning control for water-efficient agricultural irrigation. In: IEEE International Symposium on Parallel and Distributed Processing with Applications and 2017 IEEE International Conference on Ubiquitous Computing and Communications (ISPA/IUCC), pp 1334–1341. https://doi.org/10.1109/ISPA/IUCC.2017.00203
98. Vispute SR, Kolekar V, Bhujbal H, Gudle M, Kadam A, Kadu K (2018) An application for e-marketing of agricultural commodities and analysis of marketable surplus. Int J Adv Res Comp Comm Eng 7(4):77–79
99. Sutton R, Barto A (2018) Introduction to reinforcement learning, 2nd edn. The MIT Press, Cambridge MA
100. Gawali P, Dalvi M, Dhannawat D, Deshmukh S, Vispute SR (2020) An application of data analytics in agriculture sector for multi-advice generator in native language. J Critical Rev 7(19):2389–2394
101. Waghmare H, Kokare R, Dandawate Y (2016) Detection and classification of diseases of Grape plant using opposite colour Local Binary Pattern feature and machine learning for automated decision support system. In: 3rd International Conference on Signal Processing and Integrated Networks (SPIN), pp 513–518. https://doi.org/10.1109/SPIN.2016.7566749
102. Elavarasan D, Vincent PMD (2020) Crop yield prediction using deep reinforcement learning model for sustainable agrarian applications. IEEE Access 8:86886–86901. https://doi.org/10.1109/ACCESS.2020.2992480

Advancement and Evolution of LNG Terminals: A Review

Rishi Dewan, Umang Kumar Yadav, Anand Nayyar, Nikhil Verma, and Adarsh Kumar

Abstract It has been more than two decades, natural gas (NG) market is booming and being utilized on priority basis a viable energy resource among various regions across the world due to proven, low-greenhouse emission during various operation and generation of power with high efficiency. Pipelines were majorly used for safe and reliable transport of natural gas produced for almost past 100 years; pipelines proved to be an economic, stable, and secure way for supply from easily accessible locations of conventional gas reservoirs, but the major challenge comes amid the transportation of natural gas produced from the inconveniently located gas reserves. Exploiting such remotely located gas reserves could be challenging, and one such commercialized way is liquefaction of natural gas for transporting it as liquefied natural gas (LNG) to LNG terminals located at various ports. This paper reviews the new technologies adapted for evolution of LNG terminals, in terms of inherently safe process designing, optimization, automation of the facility or shifting to FLNG projects in order to increase cost effectiveness of project, increase safety during operation and reduce the complexity of the LNG market, though increasing the competitiveness for aligning the business requirements. A planned layout with leveraged automated controls can prove to be more effective in long haul operation with increased profitability and reduce the substantial hazards associated by tackling multi-dimensional

R. Dewan
Department of Physics, University of Petroleum and Energy Studies, Dehradun 248007, Uttarakhand, India

U. K. Yadav · N. Verma
Department of HSE, University of Petroleum and Energy Studies, Dehradun 248007, Uttarakhand, India

A. Nayyar (✉)
Graduate School, Faculty of Information Technology, Duy Tan University, Da Nang 550000, Vietnam
e-mail: anandnayyar@duytan.edu.vn

A. Kumar
Department of Computer Science, University of Petroleum and Energy Studies, Dehradun 248007, Uttarakhand, India
e-mail: adarsh.kumar@ddn.upes.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_57

safety issues. The integrity of LNG terminal (and whole facility) is dependent on the processes undergoing (liquefaction, transportation, storage and regasification) and compliance with the statutory requirements (NFPA 59 A). Automation will lead to reduced operating costs along with maintenance costs, adequate attention to plant safety, reliable operation, and process optimization with maximum asset utilization. The evolution of LNG market is must which will enhance security of global gas supply for consuming nations with reduced constraints (both political and geopolitical).

Keywords Inherent safety · Automation · LNG · LNG terminal · FLNG

1 Introduction

Natural gas (NG) has the potential to be the most abundant source of energy across the globe but still, almost one-third of conventional reserves remains unexploited [1]. As oil prices are at its peak with diminishing resources, there is urgent need for commercialization of technologies for exploitation of such stranded reserves of natural gas (NG) globally for safe transporting of NG over long distances, which ensures less GHG emissions upon utilization and help in meeting the expectations of growing requirement of energy [2, 3]. Transportation of natural gas (NG) becomes competitive via pipeline for distances more than 700 miles (offshore) and 2200 miles (onshore); for short distances, pipelines are more convenient and economical [4]. LNG is the one form of natural gas (NG) which can be utilized for transporting across distant location through oceans with use of LNG carriers; as construction of underwater large diameter pipeline network will be more costly and inconvenient due to requirement of gas compressions/recompressions system at suitable locations facilities to overcome the transmission pressure drop [4, 5]. Figure 1 explains the various methods available for transportation of natural gas [6]. LNG is the liquefied form of natural gas (NG) cooled at −162 °C at atmospheric pressure leading to a reduction in volume by 600 times, thus making an colorless, nontoxic, non-corrosive and odorless liquid fuel more feasible, and economic to transport in large quantity. It promises security of supply with flexibility in operation and generates less carbon

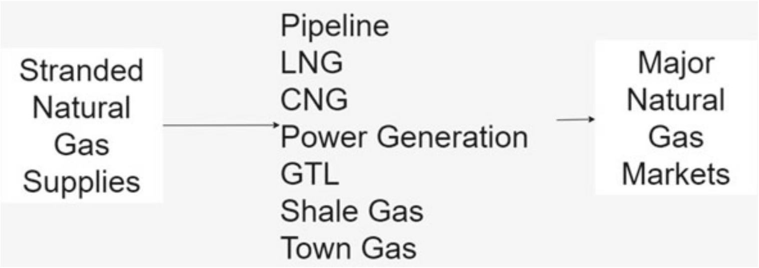


Fig. 1 Methodology available for transportation of natural gas

emission and particulate emissions making LNG a cleaner source of energy. LNG composition predominantly consists of methane and release of natural gas (NG) from LNG can lead to asphyxiation in a confined space and may ignite if mixed with adequate amount of air. Globally, the natural gas (NG) consumption is expected to rise by 1.3% per year on average basis (3.8 trillion m3 in 2018 to 5.2 trillion m3 in 2040) [7]. In coming years, LNG consumption is going to increase significantly by 22% and 17% in Europe and India, respectively, (approximation) [8]. As per 2019 review, India is 4th largest importer of LNG (31 bcm) and the 14th largest consumer of LNG (58 bcm) with an compound annual growth rate (CAGR) of 7%, boosting the economy along with sustainable energy consumption [9]. In India, LNG existing terminal is situated at Dahej, Hazira, Dabhol, Kochi, and Ennore (IOCL) with capacity of 17.5 MMTPA, 5.0 MMTPA, 5.0 MMTPA, 5.0 MMTPA, 5.0 MMTPA each, respectively, (total 37.5 MMTPA). Several LNG terminal is under construction and few proposed which will add 34 MMTPA to the total capacity, making a total of 71.5 MMTPA [10]. The demand of LNG is increasing at fast pace as the energy demand is exceeding the supply, thus Government of India plans on significantly boosting the India Energy basket by increasing natural gas (NG) share, increasing dependency on imported gas and extending the pipeline grid network (in order to connect to new markets) by 26,000 km along with formalization of gas policy for classification of the users [11]. There are 180 terminals in design stage and construction stage waiting to be operational worldwide [12]. The significant development of supply chain in LNG production site, gas treatment, transportation, liquefaction, etc., has opened the market to new opportunities to adopt new technologies that follow the inherent safety concept for evolution of the LNG terminals both onshore and FLNG projects; also several technological innovations are proposed for the LNG carries and regasification facility [12, 13]. With depletion of conventional onshore energy reserves, now focus has shifted on exploitation of reserves located offshore; the high-production rate has generated demand of setting up of cost-effective offshore floating liquefied natural gas (FLNG) facilities with combination of operations (production, storage, unloading) for ease in long haul transportation with use of newly developed ship to ship unloading facility, which significantly cuts the cost of pipeline network and establishment of onshore terminals [14, 15]. The risk arising in FLNG facility is majorly due to its complex and compact structure; the size of storage facility, oceanic waves, collisions, spills, loss of containment, operating pressure, cryogenic temperature operation, and LNG flammability [16]. The possible way to eliminate the risks is implementation of inherent safety concept in designing process which focuses on the integrity of the facility and improves safety standards [14, 17]. The goal being to design a intrinsically safe terminal covering the scope of both liquefaction and regasification facility in order to minimize the accidents or injuries associated with the operation [12, 18]. It has been observed that attaining absolute security and safety is quite difficult, just like the affordability due inevitability of statutory requirements [12]. In order to enhance the energy efficiency of the liquefaction operation, recent enhancements are there in the liquefaction cycle and driver cycle which increase the efficiency of gas turbine, reduced compressor power, and fuel consumption to generate the power of required amount with enhanced integrated gas combined cycle (IGCC); innovations are there for treatment of the sour gas indeed [19]. For the regasification operation developments and advancements are in place which will impact the overall efficiency of LNG supply chain as cryogenic power generation is witnessed and ambient air vaporizers are used instead of combustion vaporizers for implementing an ecofriendly atmosphere [20, 21]. Use of dynamic simulation has paced in the LNG industry as it directly triggers the operability and profitability of a plant during early stage of plant design and eliminate any costly work that may be required in coming future; thus improving plant design [21, 22]. Plant automation is the most modern aspect for rapid progress; from manual operation to pneumatic control system to electronic system, enhancements has been in place with use of algorithms, PID controls, sensor technology, distributed control system (DCS) and microcomputers [21]. The future of LNG facility will be decided by use of advanced process control (APC) for asset management and asset optimization using conceptual collaborative process for delivering information [21, 23]. With evolution of the LNG network, risks may arise due loss of containment during any of the stage (production, liquefaction, transportation, storage, and regasification) [24, 25]; safety issues and health issues will be prominent to the population and in order to have social acceptability the risk to the people, property, and environment should be minimized.

2 Overview

The operation undergoing onsite can be majorly classified in stages as LNG receiving/unloading from the ships, onsite storage of LNG, liquefaction/regasification/compression, and distribution of natural gas (NG). Figure 2 represents the layout of LNG terminal. The LNG from the carrier is transferred to the terminal at a temp of −162 °C via pipelines and stored at −162 °C temperature in the specially designed cryogenic tanks provided with insulation to avoid boil off. Instead to such precautions, there are some vapors released which are transferred back with use of compressor and re-condenser. The unloading occurs at the jetty (Berth) consisting of 4 unloading arms (3 are used for transferring the LNG, and 1 is used for vapor return); this is just as equalization of pressure is important to avoid any catastrophic failure. Complete movement of arms is possible in all planes. The emergency release system (ERS) disconnects the supply in case of any emergency like fire or disastrous situation. ESD 1 deals with shutdown of cargo pumps, and loading valve is closed; ESD 2 deals with safe breaking of powered emergency release coupling (PERC) to prevent any damage to structure [25]. Regasification is carried with help heat exchangers, and it involves warming up the LNG with use of air, heated water, seawater, or fired heaters in controlled environment condition; LNG is pressurized at 70 to 100 bar, above 0 °C temperature to convert it to natural gas (NG) which is odorized with tetrahydrothiophene (THT) to detect any leak and transferred to the transmission and distribution system.

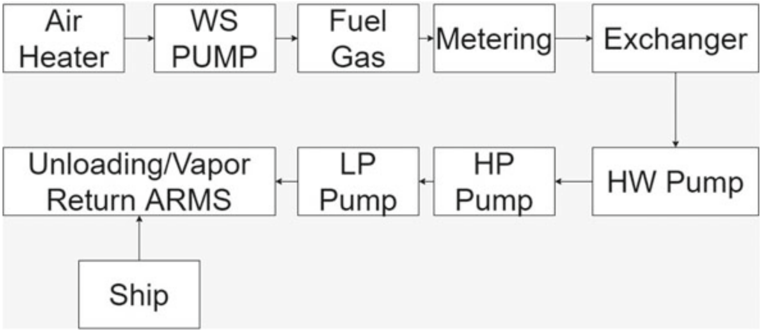


Fig. 2 Typical layout of LNG terminal [11]

2.1 LNG Terminal Located Onshore

The LNG terminal is connected worldwide with a network of LNG carrier, responsible for delivering the LNG to various terminals, which then is converted to gaseous state by regasification; and natural gas (NG) is delivered to various customers by the help of pipeline network or utilized of power generation. In most of cases, the onshore terminals are located close to industrial area and populated area in order to increase feasibility of supply but land acquisition with easy marine access is quite difficult that to close to an industrial/populated area. Moreover, there are many environmental/safety/security constraints while constructing an LNG terminal; adequate planning with compliance to regulatory standard is a must which may be time consuming and costly. LNG is unloaded from the carrier with help of pumps connected directly with unloading arm and transferred to the required storage space. The LNG from there is moved with pressure higher than the previous values and warmed up with help of air, heated water, seawater, or fired heaters in controlled environment condition. While planning, adequate spacing can be provided between equipment/operations to avoid the escalation of hazards arising due to domino effect, fire, and cryogenic protection can be provided to equipment/processes without compromising safety and focus should be on reducing the complexity of operation and the congestion caused due to network of pipelines and process equipment. If any incident occurs the safety of workers is ensured, as they can be easily located in safe areas, the major risk is beyond plant boundaries. Developments are carried out, with modular construction of structures in which site area is minimized to reduce the impact on environment, withstand extreme weather conditions and tackle situation of labor mobilization [26].

2.2 LNG Terminal Located Offshore

The difficulties arising in the onshore terminal can be tackled, with use of offshore LNG terminals providing many advantages but at a cost of increased complexity, risks, and feasibility of operation. Offshore facility has arrangement of equipment and process in close proximity due to high costs involved in construction of structural platform; fire and cryogenic protection are not only limited to a certain process or equipment, but are related to the whole facility. Blast walls, safe refuge to workers, and egress protection have to provide on high-priority basis, with use of modular designs concept [26]. Technological advancements are in place, boosting the economic constraints, reducing the risks involved, and maintain safe working environment. LNG terminals on shore are shown in Fig. 3. In offshore LNG terminal, the either production of NG is carried out on site which is liquefied and stored for transportation or is delivered by the LNG carrier and is distributed to the customer after regasification. The offshore terminals can be classified as GBS (gravity-based structures; fixed structures made out of concrete on the sea floor installed with regasification facility and storage tanks for LNG) and FSRU (floating, storage, and regasification units; mobile floating carrier installed with a mooring system or connected via jetty system). The selection of design depends upon the feasibility, site conditions, send out requirements, costs involved for construction and environmental impact.

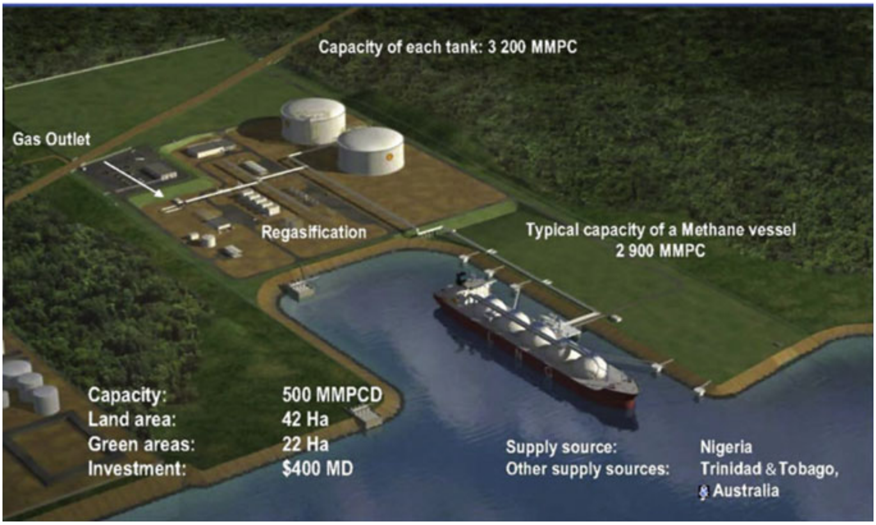

Capacity of each tank: 3 200 MMPC
Gas Outlet
Regasification
Typical capacity of a Methane vessel 2 900 MMPC
Capacity: 500 MMPCD
Land area: 42 Ha
Green areas: 22 Ha
Investment: $400 MD
Supply source: Nigeria
Other supply sources: Trinidad & Tobago, Australia

Fig. 3 LNG terminal located onshore [21]

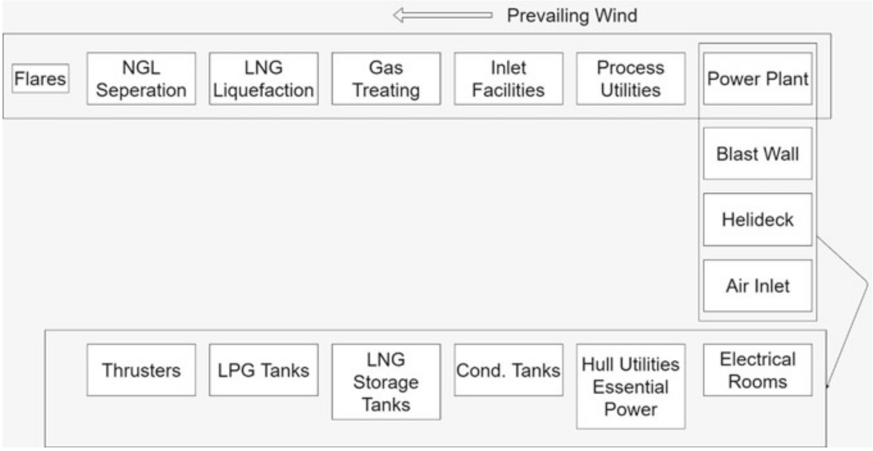


Fig. 4 Typical layout of FLNG facility [21]

2.3 Floating Liquefied Natural Gas (FLNG)

For any FLNG facility, the designing process is of utmost importance, as layout of the facility directly or indirectly decides the capital costs involved, applicability of inherent safety concept and the hazardous incidents that may arise (Fig. 4) [14]. The offshore facility requires more advanced state-of-the-art tech as compared to the onshore facility in order to fulfill the challenging objectives [27]. The major process involved is the production, liquefaction, regasification, and unloading/offloading, in which liquefaction is done with use of dual mix refrigerant, thus enhancing the efficiency and production capacity with low-capital costs involved; this technique is used by the Prelude operations undergoing at South Korea owned by Shell Global Group [28]. With innovation, comes the hazard and the most effective way to prevent, control, and minimize the hazard is the use of integrated inherent safety index (I2SI) for layout improvement and enhance safety [29].

3 Inherently Safe Approach

It is a proactive concept focusing on the elimination of hazard intrinsically during conceptual design stage of any operation [30]. It gained more attention after the explosion occurred at Flixborough in 1974, but other techniques like HAZOP and quantitative risk assessment (QRA) overcome this concept, despite of the fact that use of this concept provides with economic benefits (due to cost reduction) and prevention and elimination of hazards arising from the structure of facility [31, 32]. QRA is for analyzing the typical risks and various insecure events. It gives quantifiable way to measure the risk related to those parties which are having interests in these potential area. QRA process is perform in four major steps. The first step is hazard identification, after that the discussion on the consequence and effect analysis. In third step, there is probability and frequency estimations, and in fourth stage, the risk calculation involves which leads to a risk analysis. This analysis is quantitative in nature and it might outcome against any form of risk assessments or risk acceptances, process, and systems. Since the inception of this concept, several indices like integrated inherent safety index (I2SI) and safety weighted hazard index (SWeHI) have been developed, which quantifies the risk involved, and provide with a more accurate risk evaluation result. This approach has been used successfully in various oil and gas (O&G) facilities and other major industrial operations [29]. The suitability and basis of principles involved in inherent safety concept are defined by the use of the specific key words; all of which cannot be applied in the initial stage of designing. The most dominating guide words while designing in early stage of any process were attenuation (alteration in the arrangement of the units in order to avoid hazards arising from domino effect), simplification (designing process simplification, pipeline network complexity simplification), and limitation (areas affected limitation, domino hazard escalation limitation, damage limitation significant to the structures). Some of the above keywords are implemented as principles; also, hence, the principles should be the focal point while designing the inherently safe terminal. The 4 principles are:

1. Minimization: Control over the hazardous material inventory, reduction in all ways possible to minimize the hazards
2. Substitution: Replacing the more hazardous material either with less hazardous material or safer material, like use of water instead of heating oil
3. Attenuation: Using hazardous material in least hazardous, like storing cryogenic liquid at atmospheric pressure not at ambient temperature under pressure.
4. Simplification: Reduce complexity of process, reduce hazards by reducing equipment.

4 Layout Optimization

The comprehensive way for layout optimization in order to ensure inherent safety concept assessment is use of domino hazard index (DHI) and I2SI, and it helps in connecting the cost involved and the inherent safety aspect. Layout optimization focuses on enhancing the current arrangement/design in order to increase the safety to much higher extent quantitatively; the area mostly prone to hazard is the process area, where multiple process is undergoing side by side under varying condition of temperature and pressure. Assessment of the whole facility is must in order to eliminate the possibility of any hazards occurring, but major focus should be on the process area as passive measure is more prominent there like the liquefaction process carried out at -162 °C; threatening the FLNG facility [33]. The failure to use the passive protecting measures like the fire walls or blast walls significantly increases the possibility of hazards arising due to domino effect. Yuchen Wang able to explain the optimization of berth length of LNG terminal [35]. The mixed-integer linear programming (MILP) method is used to demonstrate the problem. In MILP approach, there are few variables which are constrained to be integers and rest another variables are allowed to be non-integers. The MILP model of operational optimization of LNG terminal is one of the advance way of layout optimization [36].

5 LNG Terminals and Associated Insecurities

LNG is consider in a fuel categories which is pure in form, environment friendly, and the most utilized energy source worldwide. The hazards associated are:

- Spilled LNG
- Jet fire
- Pool fire
- Vapor cloud explosion (VCE)
- Cryogenic effects
- Rollover conditions in storage tanks
- Confined space
- Metal embrittlement
- Chemical hazards.

6 Safety Associated with LNG Terminals

In order to ensure safety of the facility, the hazards associated have to be eliminated. The escalation of hazards mentioned above has to be controlled along with: compliance with codes/standards/regulations that are accepted national and internationally for designing terminals.

- Use of intrinsically safe system; use of suitable material of insulation of LNG to render any problems arising during storage.
- Use appropriate risk assessments tools for deciding location of LNG terminals with respect to populated area.
- Minimizing consequences arising due to pool fire escalation, containing spills by construction of impoundment areas.
- Use of emergency release system (ERS) and powered emergency release coupling (PERC) incorporated with the unloading arms to avoid any hazardous event arising due to deviation from acceptable parameter.
- Temperature regulator to regulate the temperature conditions in various processes across the facility.
- Pressure regulator and relief valves to overcome issues rising due to overpressure.
- Leak detection devices and spill control using probes.
- Fire and combustible vapor detection devices.
- Hazardous area classification.

- Automatic emergency shutdown system.
- Automatic depressurization system.
- Active and passive fire protection.
- Training to operators for quick response in case of emergency.
- Timely maintenance of equipment.

7 Statutory Regulation

The organization that is in charge of overseeing and exercising control over the petroleum industry is called the Petroleum and Natural Gas Regulatory Board. Its mission is to monitor and regulate the petroleum industry. The act provides for the establishment of the Petroleum and Natural Gas Regulatory Board in order to protect the interests of consumers and entities engaged in certain activities relating to petroleum, petroleum products, and natural gas, as well as to promote competitive markets and for matters that are connected therewith or incidental thereto. Additionally, the act develops for the establishment of the board provisions for matters that are connected therewith or incidental thereto. This is done to encourage competitive markets and for other things that are linked to or incidental to the promotion of competitive markets. In addition, the act includes provisions covering items that are connected to or incidental to the subject matter of the act itself. These provisions are referred to as "related matters." The mandatory statutory limits that must be adhered to throughout the development, construction, and operation of LNG facilities are outlined in Table 1, which offers an overview of these requirements [34]. Further as enshrined in the act, the board has also been mandated to regulate processing, storage, transportation, distribution, marketing, and sale of petro natural gas.

8 LNG Terminal Automation and Advancement

The prime objectives for automation of LNG terminals are.

- Ensure overall safety
- Ensure stability of operations
- Utilizing benefit of advanced process control (APC)
- Optimization of asset operations
- Optimization of asset performance.

Ensuring safety is of prime importance while upgrading any terminal, all the safe operating limits must be feed accurately to ensure feasibility of automation, which can be achieved with use of base regulatory loops and advanced regulatory loops. The DCS is connected to the alarms, which warn the operators about any mishappening and alert them to take recommended action; the SIS ensures that plant is securely operational; and shutdown during abnormal conditions shutdown. For

Table 1 Statutory regulations mandated for design, construction, and operation of LNG terminal [34]

S. no	Statutory Regulation	Overview
1	NFPA 59A	"Regulatory standard for handling of LNG (including production and storage of LNG liquefied natural gas) 2006"—Following is the part of NEPA 59A • LNG facility • Vaporization facilities • Piping and instrumentation system component • Process systems • Storage containers for LNG • Natural gas transportation, and refrigeration, • Protection from unexpected fire and similar insecurities, • LNG and its supply chain security
2	EN1473	"Installation and equipment for liquefied natural gas design of onshore installations"—This includes an risk-based approach for designing the LNG terminals. This standard was evolved from the BS 777,742 in 1996
3	EN1160	"Installation and equipment for liquefied natural gas general characteristics of liquefied natural gas"—This includes guidelines over the features of substances prominent in LNG terminals
4	49CFR Part 193	"Liquefied natural gas facilities"—This includes • Designing aspect • Siting requirement • Construction methodology • Equipment used and specification • Operations undergoing • Maintenance of equipment • Personnel qualification • Training of personnel • Fire protection • Safety of LNG and security of supply
5	33CFR Part 127	"Waterfront facilities handling liquefied natural gas and liquefied hazardous gas"—This includes factors governing import and export of LNG
6	33CFR 160.101	"Ports and waterways safety"; "control of vessel and facility operations"—These showcase the duties and responsibilities practiced by the commanders appointed at higher position in district and Port Captain in order to confirm the security of infrastructure which vessels and waterfront facilities. The controls are in direction to specific hazards and incident only
7	33CFR 165.20	"Regulated navigation areas and limited access areas"; "safety zones"—These include • Environmental aspect • Safety • Authorized persons, vehicles, or vessels
8	33CFR 165.30	"Regulated navigation areas and limited access area"; "security zones"—These showcase area of land/water as security zone designated by District Commander or port, harbors, and territories from sabotage

(continued)

Table 1 (continued)

S. no	Statutory Regulation	Overview
9	EEMUA 14,743	"Recommendations for the design and construction of cryogenic LNG storage tanks"—This includes suggestions and initiative for the design and building infrastructure. This infrastructure may include containment tanks which are having single, double, or full storage capacities for LNG at cryogenic temperature

smooth operation can be achieved with use of base regulatory controls, tuning up the control loops and use of advanced process and regulatory controls for enhancing operational performance. The optimization of the asset operations can be achieved by use of real time optimization or APC, which considers the market factor of boosting economic viability. The management and maintenance of asset are must to ensure the overall optimization of the plant facility to boost its performance in long haul. The advanced process control (APC) helps in running the terminal at optimal conditions with maximum output generation at minimal capital investment, which will provide with many benefits for ant LNG terminal as the efficiency of operations will be enhanced with improved product quality, and thus, it will lead to maximized profit to operators.

In order to have promising results from the facility, certain innovations and advancements are mandatory in varied areas of LNG terminal operation, equipment, and process.

- Innovation in liquefaction and regasification operation
- Use of large train size
- Change in size and capacity of main heat exchanger
- Liquefaction pressure adjustment
- Integrated natural gas liquid (NGL) recovery and LNG liquefaction process
- Integrated LNG regasification and power generation unit
- Efficient liquid expander and ambient air vaporizer
- Efficient gas turbines and electric motor drive
- LNG Wobbe index
- CO_2 emission reduction
- Controlled emission from ship propulsion system
- Enhanced integrated gas combined cycle (IGGC) plant
- Modularization
- Improved design for LNG berthing facility
- Matching drives and colder climate design of facility.

9 Flaring System

In order to ensure stability of the operation undergoing in LNG facility, the process unit is fabricated with control system, which makes sure that the equipment are operating under the expected design temperature and design pressure. Whenever plant is undergoing start up procedure, emergency situations may arise due to abnormal condition build up, which directed flaring system to confirm safety of plant. Relief and flare system design should be carried out ensure sustained operation within the facility, which can be done by accommodating the maximum relief loads arising from the process units during off-design or emergency situations. Certain engineering documents (process unit/utility system heat and material balance datasheet, power distribution diagram, flare distribution diagram, and plant layout with equipment location) are required and must be analyzed for designing the flaring system of any facility [37].

Objectives of flaring system are

- Minimize discharge in the atmosphere
- Provide safe working environment to the personnel, ensure safety of workers and equipment
- Comply with safety norms, regulations, standards, and design code.
- Reduction in impact of relief on the environment
- Recovering boil off gas/vapor/liquid for economic viability
- Reduction in emissions, flare noise, flare smoke, and flare luminance to minimize impact on community.

10 Conclusion

To meet the futuristic energy demand, the LNG plant and associated infrastructure are pushing up the production capacity by twice; the efficiency of both liquefaction and regasification terminals has increased with promising results. The marine carriers have increased capacity to 250,000 m^3 storage tanks for transporting LNG. The trains size is modified for the coming era of evolution, and it is justified that train size ranging from three to six MTPA will be in more use as most of the LNG terminals will be shifted offshore due to flexibility of operation, less environmental constraints, more efficiency, increased output, and rewarding operating and capital cost. The application of advanced process control (APC) in today's era is less but is promised to increase in coming future due to changing market condition and energy demand tend to increase in coming years, thus pushing the operator to move to economically viable and beneficial options to maximize the terminal potential; with this the quality of LNG/NG has to be as per the requirements of market and must be tailored on time to time basis. Modularization is the growing concept which can be used in the construction phase of LNG terminal at remote location. It makes use of state-of-the-art self-supporting transportable design structures consisting the equipment and piping all together, transported to site from the fabrication unit. Due to complexity of the LNG facility, the concept of modularization will have difficulty in implementation, but with innovation and development, the cost effective modularization can be implemented in coming years. Effectively designed maintained leveraged control system is responsible for reducing startup time. It is also responsible for maintaining optimum operating profit. With all the advancements, the LNG terminals have no doubt evolved, but safety should still be of paramount importance, the hazards arising due to domino effect should be minimized, project execution should be in hand of experienced personnel who could suitably maintain the health condition, ensure safety, keep in mind the environmental constraints, ensure reliability of operation, and ensure long life of the plant with timely maintenance. The Next Gen LNG terminals will be focused on reducing the impact of emissions on environment, thus providing with less carbon footprint with increased energy efficiency by use of high-efficiency gas turbines, IGGC plants, and a more efficient NGL integration.

References

1. Thackeray F, Leckie G (2002) Stranded gas: a vital resource. Petroleum Econ (English edition) 69(5):10–12
2. Mokhatab S, Wood D (2007) Breaking the offshore LNG stalemate. World Oil **228**(4)
3. Wood D, Mokhatab S (2008) Commercial breakthroughs in LNG technology. World Oil 229(10):135–140
4. Mokhatab S, Purewal S (2006) Letter to the editor: Is LNG a competitive source of natural gas? Pet Sci Technol 24(2):243–245
5. Mokhatab S et al (2014) LNG fundamentals. Handbook Liq Nat Gas 1:1–106
6. Wood D, Mokhatab S (2008) Gas monetization technologies remain tantalizingly on the brink. World Oil 229(1):103–108
7. Sieminski A (2014) International energy outlook. Energy Info Admin (EIA) 18
8. Licari FA, Weimer CD (2011) Risk-based siting considerations for LNG terminals–Comparative perspectives of United States & Europe. J Loss Prev Process Ind 24(6):736–752
9. Kadam S, Kar SK (2019) Energy security & sustainability: role of natural gas in Indian context. PDPU J Energy Manag 3(2):37–49
10. Mehrotra A, Gupta A (2020) Indian gas market—roadmap for creation of an efficient gas market. In: Energy, Environment and Globalization. Springer, pp 95–115
11. Renjith V, Kumar PH, Madhavan D (2018) Fuzzy FMECA (failure mode effect and criticality analysis) of LNG storage facility. J Loss Prev Process Ind 56:537–547
12. Felix F, Durr C, de la Vega F, Designing safety into LNG export/import plans incorporation de la securite dans la conception des usines d'exportation/importation de GNL
13. Eurostat and U.e.C. européenne (2011) Energy, transport and environment indicators, vol 2. Office for Official Publications of the European Communities
14. Xin P, Ahmed S, Khan F (2015) Inherent safety aspects for layout design of a floating LNG facility. In: ASME 2015 34th International Conference on Ocean, Offshore and Arctic Engineering. American Society of Mechanical Engineers Digital Collection
15. Won W et al (2014) Current trends for the floating liquefied natural gas (FLNG) technologies. Korean J Chem Eng 31(5):732–743

16. Pitblado RM, Woodward JL (2011) Highlights of LNG risk technology. J Loss Prev Process Ind 24(6):827–836
17. Kletz TA (2003) Inherently safer design—its scope and future. Process Saf Environ Prot 81(6):401–405
18. Martin MW, Schinzinger R (1989) Ethics in engineering. McGraw-Hill
19. Townsend D, Linnhoff B (1983) Heat and power networks in process design. I: Criteria for placement of heat engines and heat pumps in process networks. II: Design procedure for equipment selection and process matching. AIChE J 29(5):742–771
20. Pham TN et al (2017) Enhancement of single mixed refrigerant natural gas liquefaction process through process knowledge inspired optimization and modification. Appl Therm Eng 110:1230–1239
21. Mokhatab S, et al (2013) Handbook of liquefied natural gas. Gulf Professional Publishing
22. Agachi PS, et al (2007) Model based control: case studies in process engineering. Wiley
23. Ott CM, et al (2015) Large LNG trains: technology advances to address market challenges. Gastech. Singapore
24. Paltrinieri N, Tugnoli A, Cozzani V (2015) Hazard identification for innovative LNG regasification technologies. Reliab Eng Syst Saf 137:18–28
25. George JJ et al (2019) Application of fuzzy failure mode effect and criticality analysis on unloading facility of LNG terminal. J Loss Prev Process Ind 61:104–113
26. Tanabe M, Miyake A (2010) Safety design approach for onshore modularized LNG liquefaction plant. J Loss Prev Process Ind 23(4):507–514
27. Hwang J-H, Roh M-I, Lee K-Y (2013) Determination of the optimal operating conditions of the dual mixed refrigerant cycle for the LNG FPSO topside liquefaction process. Comput Chem Eng 49:25–36
28. Nibbelke R, Kauffman S, Pek B (2002) Double mixed refrigerant LNG process provides viable alternative for tropical conditions. Oil Gas J 100(27):64–64
29. Khan FI, Amyotte PR (2002) Inherent safety in offshore oil and gas activities: a review of the present status and future directions. J Loss Prev Process Ind 15(4):279–289
30. Hansson SO (2010) Promoting inherent safety. Process Saf Environ Prot 88(3):168–172
31. Lutz WK (1997) Advancing inherent safety into methodology. Process Saf Prog 16(2):86–88
32. Kletz TA (1999) The constraints on inherently safer design and other innovations. Process Saf Prog 18(1):64–69
33. Suardin J, Mannan MS, El-Halwagi M (2007) The integration of Dow's fire and explosion index (F&EI) into process design and optimization to achieve inherently safer design. J Loss Prev Process Ind 20(1):79–90
34. Alderman JA (2005) Introduction to LNG safety. Process Saf Prog 24(3):144–151
35. Wang Y (2021) Optimization of berth length of LNG terminal. E3S Web of Conferences, vol. 248, p. 03009. EDP Sciences
36. Ye Z, Mo X, Zhao L (2021) MINLP model for operational optimization of LNG terminals. Processes 9(4):599
37. Choi S, Sourirajan V, Li G (2016) High pressure relief systems in LNG receiving terminals–a safety case for HP flare. In: Mary K O'Connor Process Safety Symposium. Proceedings 2016, Mary Kay O'Connor Process Safety Center

Prediction and Analysis of Cardiovascular Disease Using Multivariate Logistics Regression Classification

Tanishka Mohan, Gulrej Ahmed, Hemlata Goyal, and Bhupendra Singh Kirar

Abstract Cardiovascular diseases are related to blood vessels and heart disorders. Throughout the world, millions of people died every year due to cardiovascular diseases. Therefore, proactive prediction of these diseases is required to attenuate fatal situations to a great extent. A machine learning approach is employed with logistics regression algorithm to predict the chance of heart attack caused by cardiovascular disorders. The logistics regression is diverse machine learning classification algorithm, which provides reasonable accuracy rate and is vividly utilized in the field of research of machine learning-based prediction models. The classification has been achieved with 85.8%, 93%, 90%, and 83% of prediction accuracy, AUC, precision, and recall, respectively.

Keywords Cardiovascular diseases · Cardiac arrest · Machine learning · Logistic regression

1 Introduction

The heart is very important organ of human body. It is divided into two separating pumping system, the right side and left side, the right side of the heart transfers oxygen-deficient blood and dump the carbon dioxide, whereas the left side transfers oxygen-rich blood and pumps it throughout body through arteries [1, 2]. There are various diseases that affect the heart pumping and functioning. Among various diseases, cardiovascular diseases are the most complicated ones. The common name for the CVDs is Heart Attack, which is most prevalent and considered as the most

T. Mohan · G. Ahmed (✉) · H. Goyal
Department of Computer and Communication Engineering, Manipal University Jaipur,
Jaipur-Ajmer Express Highway, Dehmi Kalan, Near GVK Toll Plaza, Jaipur, Rajasthan 303007,
India
e-mail: gulraj.ahmed@jaipur.manipal.edu

B. S. Kirar
Department of Electronics and Communication Engineering, Indian Institute of Information
Technology, Bhopal, Madhya Pradesh 462003, India

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_58

fatal. As per the "world health organization" (WHO), the CVDs is the leading diseases causing death across the world. Figure 1 shows top 10 causes of death worldwide [3]. The main purpose of this article is to predict and analyze cardiovascular diseases based on available datasets, this will provide a better idea to regular human being with easy-to-understand information on cardiovascular diseases, including risk factors. At the juncture of statistics and artificial intelligence, machine learning is one of most remarkably progressing engineering application fields of computer science [4]. Basically, heart attack is one of the major types of CVDs and mostly created by a blockage that stops blood flow to the brain and heart [5]. In supervised machine learning, the prediction is based on input variable let's say x, and mapped to an output variable y. In this kind of prediction, the input variable (x) is supposed to follow the output variable (y) by some functional association ($f(x)$), the function $f(x)$ is termed the predictive model. In supervised learning, x is not a single quantity but is composed and made by N number of different features [6].

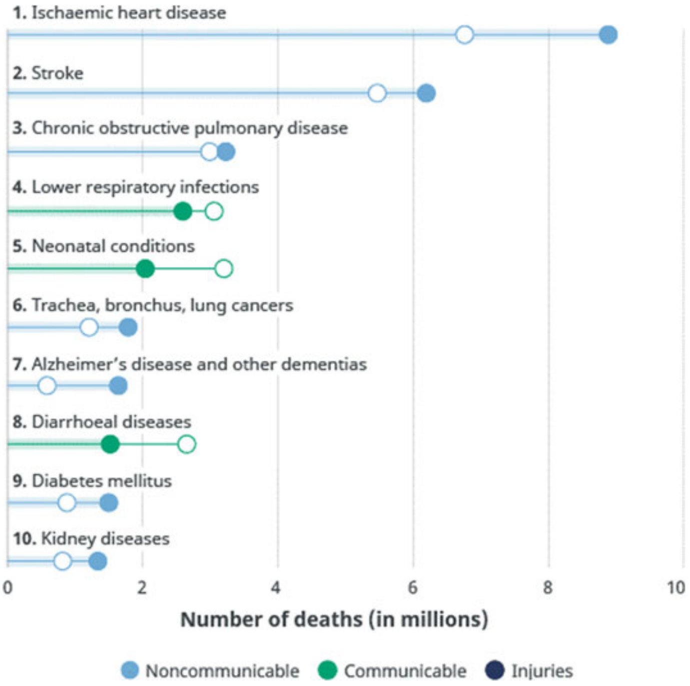


Fig. 1 Causes of death worldwide [4]. *Source* WHO Global Health Estimates

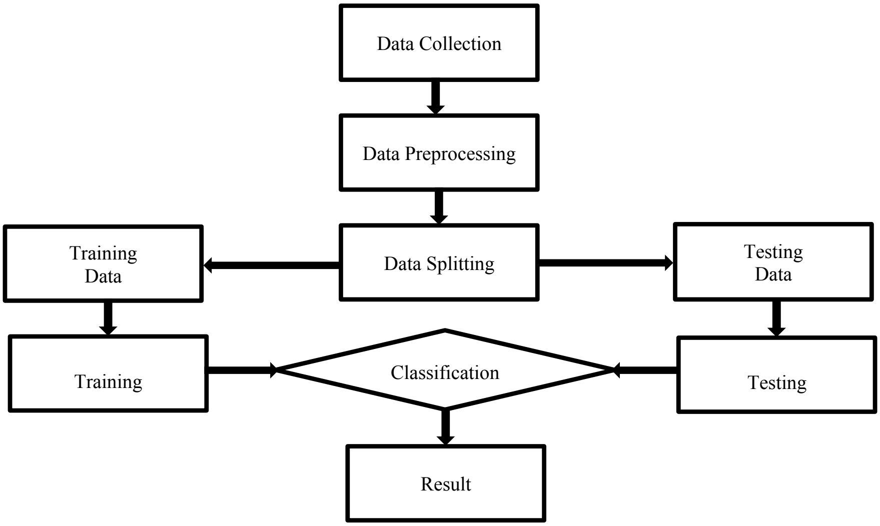


Fig. 2 Methodology for CVDs detection

2 Methodology and Framework

To predict the presence of CVDs in an individual patient, one of the major machine learning techniques termed as supervised classifications is used here. Basically, the predictive modeling of CVDs is performed with respect to classification of the dataset, where a class label of dataset is predicted for CVDs presence or absence in the patient.

In this paper, the methodology predicts the probability of CVDs a heart by using logistics regression classifier.

The chances of an individual having any kind of CVDs depends on the various attributes, which are mentioned in the section of features selection and analysis. The methodology for CVDs detection is shown in Fig. 2.

The CVD dataset has been taken from Center for Machine Learning and Intelligent Systems [7], which is basically a UCI machine learning repository.

Dataset characteristics is multivariate, it contains 11 attributes (features) and 303 instances. The details of attribute information of the dataset is given in Table 1.

3 Classification Algorithms

The logistics regression is diverse machine learning classification algorithm, which provides reasonable accuracy rate and is vividly utilized in the field of research of machine learning-based prediction models.

There are two kinds of regression algorithms namely linear and logistic regression, in a linear regression algorithm, the output is a continuous value that happens to be

Table 1 Description of heart disease dataset

S. no	Attribute	Description
(a)	cp	Types of chest pain (i) Typical (ii) Atypical (iii) Non-anginal
(b)	trestbps	Blood pressure (in mm Hg)
(c)	chol	Cholesterol
(d)	Fbs	Blood sugar
(e)	restecg	ECG results
(f)	thalach	Heart rate (maximum)
(g)	exang	Exercise induced angina
(h)	oldpeak	ST depression induced by exercise relative to rest
(i)	slope	The slope of the peak exercise ST segment
(j)	ca	Number of vessels (major 0–3)
(k)	thal	Thallium stress test result

on analog scale, which may be integer or a floating-point value. Logistic regression is basically supervised classification algorithm and utilized for categorical or binary target variables. In these supervised learning algorithms, the classification is done with the help of logistic regression algorithm.

Logistic regression algorithm is a logit function, which is basically statistical model to estimate binary values. These levels are also called categorical labels of dataset. The logit function is represented mathematically as

$$\operatorname{logit}(p) = \log_e \frac{p}{1-p} \tag{1}$$

$$\frac{p}{1-p} = e^{b_0+b_1x_1+b_2x_2+b_3x_3+\ldots+b_nx_n} \tag{2}$$

where p represents probability. x_n is nth attribute in the dataset and b_n is the nth coefficient.

4 Feature Selection and Analysis

The important features and their effect on chance of heart attack is done and presented in the following Figs. 3, 4, and 5 in the form of bar plots, it is evident that the chest pain, abnormality in ECG, the number of vessels and exercise angina directly reflects the presence of CVDS.

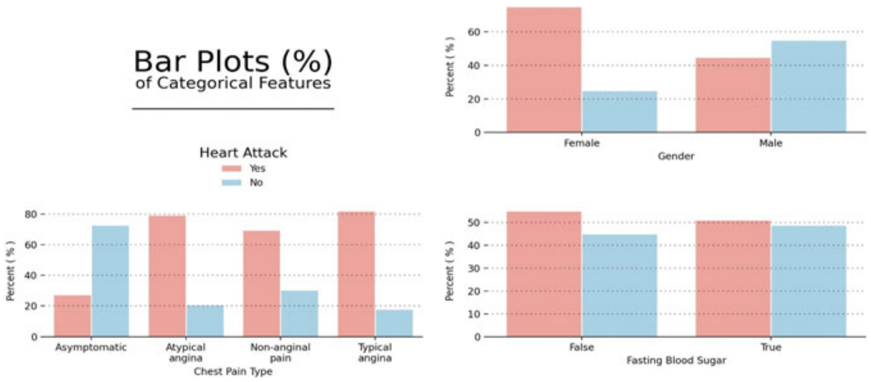


Fig. 3 Features analysis with respect to chance of CVDs as heart attack

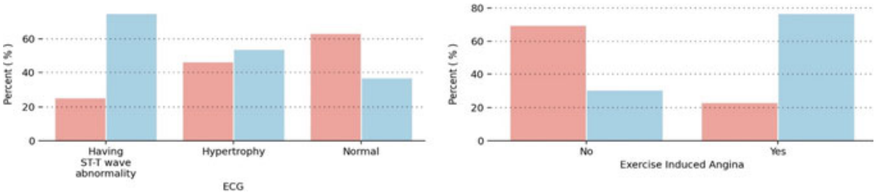


Fig. 4 ECG features analysis with respect to chance of CVDs as heart attack

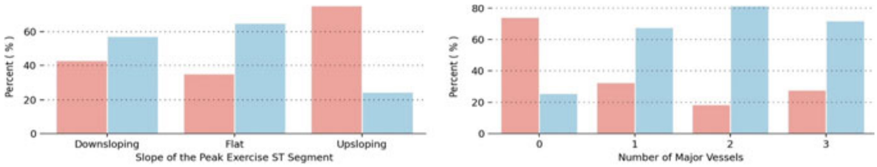


Fig. 5 Slope of peak exercise ST segment features analysis with respect to chance of CVDs as heart attack

5 Results and Discussion

In data grouping stage, the dataset has been divided into training and testing groups: The training data proportion is 80% and testing data is 20%. The results analysis of prediction of heart attack as CVDs is done with the help of confusion matrix and ROC curve, which are represented in Figs. 6 and 7, respectively. Table 2 shows Accuracy, AUC, Recall, Precision and $F1$ score for the training data.

To observe the performance of machine learning model, the various performance parameters are considered. These parameters are dependent on outcome of the classifier. These outcomes are true positive, true negative, false positive, and false negative predictions. For better understanding, they are defined and described as follows.

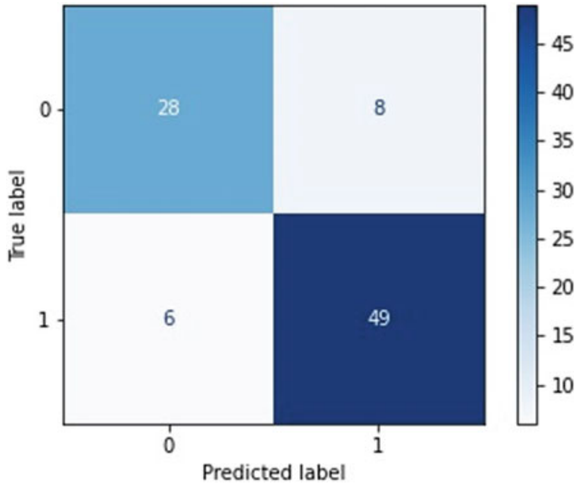


Fig. 6 Confusion matrix for predicted label

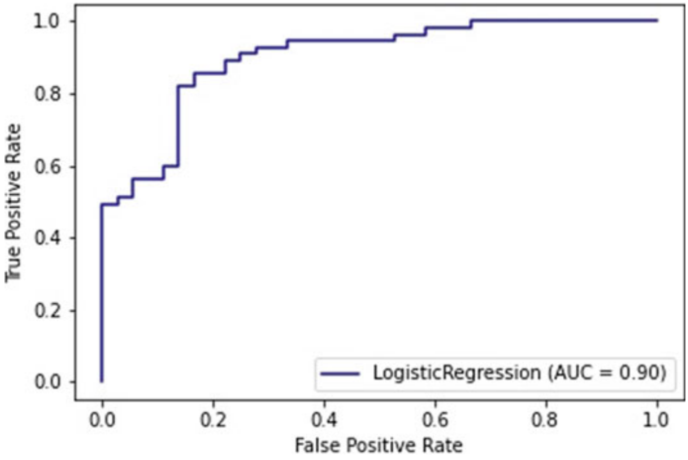


Fig. 7 ROC for true positive and false positive rate

Table 2 Result Analysis

S. no	Evaluation parameter	%
1	Accuracy	85.8
2	AUC	93
3	Recall	90
4	Precision	83
5	*F*1 score	87

- True Positive Prediction: The number of predictions of heart attacks as CVDs were perfectly matched with actual heart attack cases in the dataset.
- True Negative Prediction: The number of predictions of healthiness (no CVDs) were perfectly matched with actual cases (no CVDs) in the dataset.
- False Positive: It represents number of predictions with CVDs, which were actually not having any CVDs.
- False Negative: It represents number of predictions of no CVDs, which were actually having CVDs.

For better understanding, let assign the binary logic 0 in dataset for heathy person who were not having CVDs and binary logic 1 for not heathy person having CVDS. Then The confusion matrix interpretation.

- The True Positive Prediction (1,1)—Their actual value is 1 in the dataset and the classifier predicts it as 1.
- The True Negative Prediction (0,0)—Their actual value is 0 in the dataset and the classifier predicts it as 0.
- The False Positive Prediction (0,1)—Their actual value is 0 in the dataset and the classifier predicts it as 1.
- The False Negative Prediction (1,0)—Their actual value is 1 in the dataset and the classifier predicts it as 0.

The accuracy is defined as the fraction or ration of predictions of classifier, which were correct to the total number of pf predictions. Formally, the accuracy can be formulated as following.

- $\text{Accuracy} = \frac{\text{Number of correct prediction}}{\text{Total number of prediction made}}$
- $\text{Precision} = \frac{\text{Number of True Positive Prediction}}{\text{True positive predition+False positive prediction}}$
- $\text{Recall} = \frac{\text{Number of True Positive Prediction}}{\text{True positive predition+False negative prediction}}$
- $F1\ \text{Score} = \frac{2 * \text{Recall} * \text{Precision}}{\text{Recall} + \text{Precision}}$

The performance parameters are tabulated in Table 2 as following.

6 Conclusion

It is also observed that the implementation of machine learning-based techniques for cardiovascular disease prediction is improving the accuracy and reducing the cost factor of health expenses. The Identification of the disease plays an essential role in our busy life schedules. Particularly, the prior diagnosis of CVDs can lessen the damage to the heart and prevent heart failure. The prediction was done using logistic regression technique and the accuracy of the prediction was 85.8%. The Future works will focus on using more advanced machine learning models and better optimization techniques such as those using ANN (neural network models) and give better results accordingly.

References

1. Wernick M, Yang Y, Brankov J, Yourganov G, Strother S (2010) Machine learning in medical imaging. IEEE Signal Process Mag 27:25–38. https://doi.org/10.1109/MSP.2010.936730
2. Allarakha S, Uttekar PS, What are the four main functions of the heart?. https://www.medicinenet.com/what_are_the_four_main_functions_of_the_heart/article.htm, Accessed 17 July 2021
3. Alexopoulos EC (2010) Introduction to multivariate regression analysis. Hippokratia 14(Suppl 1):pp. 23–28. PMID: 21487487; PMCID: PMC3049417
4. The top 10 causes of death. https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death, Accessed 01 July 2021
5. Asif S, Wenhui Y, Tao Y, Jinhai S, Jin H (2021) An ensemble machine learning method for the prediction of heart disease. In: 2021 4th IEEE. International Conference on Artificial Intelligence and Big Data (ICAIBD), pp 98–103, May. https://doi.org/10.1109/ICAIBD51990.2021.9459010
6. Jordan MI, Mitchell TM (2015) Machine learning: trends, perspectives, and prospects. Science 349(6245):255–260. https://doi.org/10.1126/science.aaa8415
7. Statlog (Heart) Data Set. http://archive.ics.uci.edu/ml/datasets/Statlog+%28Heart%29, Accessed 1 July 2021

A Survey on Multilevel Thresholding-Based Image Segmentation Techniques

Saifuddin Ahmed and Anupam Biswas

Abstract Multilevel thresholding is one of the most widely used techniques for image segmentation. A thresholding technique for image segmentation is mainly categorized into two types such as bi-level and multilevel thresholding. A single threshold value is used in bi-level thresholding for image classification such as—foreground object and background object. Bi-level thresholding gives unsatisfactory segmentation results in case of complex image; hence, the idea of multilevel thresholding has been preferred over bi-level thresholding method. In multilevel thresholding, selection of threshold values mostly gives inaccurate values, and it is a time-consuming process. Hence, automatic multilevel thresholding techniques are used as an objective functions to choose optimal threshold values but faces high computational complexity problems. Meta-heuristic algorithms play an important role to reduce the computational complexity of multilevel thresholding. In this paper, we have surveyed various objective functions used in automatic multilevel thresholding and performed a comparative study about the performances of some recent meta-heuristic algorithms, which are widely used in multilevel thresholding. Also, discussed different datasets and metrics used to evaluate multilevel thresholding techniques. In addition, some applications of image segmentation are also discussed.

Keywords Parametric · Nonparametric · Meta-heuristics · Optimization

1 Introduction

The image segmentation technique is one of the vital steps in Image processing. Segmentation sub-divides an image into useful and meaningful objects or regions or extracts its boundaries. It is used to extract interested objects of an image from its

S. Ahmed (✉) · A. Biswas
Department of Computer Science and Engineering, National Institute of Technology Silchar, Cachar 788010, India
e-mail: saif783335@gmail.com

A. Biswas
e-mail: anupam@cse.nits.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_59

background. Image segmentation is considered as the preprocessing stage in higher-order processing like—object recognition, computer vision, analysis of image, and pattern recognition [1, 2]. This concept can be used in many fields such as—medicine [3]; it is used for breast cancer detection using multilevel thresholding-based segmentation technique, in agriculture [4] like detection of green plants inside a sweet corn field and in surveillance [5] like traffic monitoring, tracking crime or accidental activities, and security. Image segmentation has many applications like—detection for brain tumor, character recognition, and face recognition [6]. It has a well scope in the areas like—content-based image processing, detection of object, medical image processing, and security and surveillance [7]. Also, segmentation plays a vital role in many computer-aided diagnosis like microscopy images, ultrasound images, dermoscopy, X-ray images, computed tomographical (CT) images, and positron tomography [8].

Histogram of an image plays an important role in image segmentation [9, 10]. Histogram-based thresholding is very popular image segmentation techniques for its ease of use and effectiveness. Thresholding techniques are broadly categorized into the following types: bi-level and multilevel thresholding. In bi-level thresholding, a single threshold value is being used to subdivide an image into two classes: the object and the background. However, for multilevel thresholding, more than one threshold values are used to subdivide an image into multiple classes. Let I be an image to be segmented using a threshold, t. Then, the segmented image, I_{seg}, is represented as,

$$\begin{aligned} I_{\text{seg}}(m, n) = 0, if I(m, n) \leq t \\ 255, if I(m, n) > t \end{aligned} \tag{1}$$

Choosing an appropriate threshold value is very much difficult for the purpose of obtaining a good segmentation. A threshold can be chosen manually or automatically. Manual selection of a threshold is mostly erroneous, and time complexity is very much high. However, the threshold selection using automatic techniques is fast, accurate, and user independent. Hence, the automatic threshold selection techniques are used for image segmentation concept on a large scale. The basic concept of an automatic thresholding technique is that it uses an objective (maximization or minimization) function to select the threshold levels. Some of the commonly used objective functions for automatic thresholding techniques are Kapur's entropic method, Otsu's class variance concept, and cross-entropy-based method, minimum cross-entropy, Masi entropy, Tsallis and Renyi's entropic methods, etc. [11–13].

The concept of automatic thresholding techniques was actually used for bi-level thresholding, and later on, the idea is being used widely for multilevel thresholding. If an image contains a single object with clear background, then bi-level thresholding gives a very good qualitative segmentation. However, bi-level thresholding is not much effective for complex image that contains multiple objects. Multilevel thresholding is prevalent option for segmenting complex images. Automatic thresholding techniques are used to search for a set of optimal threshold values, where an objective function is used as a maximization or minimization function. Hence, multilevel thresholding techniques may be considered to an optimization problem. The main aim of this survey paper is to do a comparative analysis of some recent multilevel thresholding concepts for image segmentation using various meta-heuristic algorithms.

Rest of the paper is arranged as follows. We discuss about objective functions for automatic multilevel thresholding in Sect. 2, some popular multilevel thresholding techniques in Sect. 3, comparison of some recent multilevel thresholding methods in Sect. 4 and finally the conclusion and future scope in Sect. 5.

2 Objective Functions for Automatic Multilevel Thresholding

Various objective functions are considered to select the optimum threshold levels for multilevel thresholding concept for image segmentation techniques.

2.1 Kapur's Entropy

Kapur's concept [11] maximizes the entropy of segmented classes to select the optimum threshold values. The main concept of Kapur's method is based on Shannon's entropy, which is based on the notion of probability of occurrence [14]. Kapur's method considers gray-level histogram for representing the entropy of an image. Suppose there are m number of classes $C : \{0, 1, \ldots, m\}$ in respect of m thresholds t: $\{1, 2, \ldots, m\}$, then Kapur's entropic concept is represented as follows-

$$J_{kp}(t : \{0, 1, \ldots., m - 1\}) = H : \{1, 2, \ldots, m + 1\} \tag{2}$$

where

$$H_1 = -\sum_{j=0}^{t_0-1}\left({}^{p_i}\!/_{\omega_0}\right), \omega_1 = \sum_{j=0}^{t_0-1} P_i$$

$$H_2 = -\sum_{j=t_0}^{t_1-1}\left({}^{p_i}\!/_{\omega_1}\right), \omega_2 = \sum_{j=t_0}^{t_1-1} P_i$$

$$H_{m+1} = -\sum_{j=t_{m-1}}^{l-1}\left({}^{p_i}\!/_{\omega_m}\right), \omega_{m+1} = \sum_{j=t_{m-1}}^{l-1} P_i$$

Here, $H : \{1, 2, \ldots, m + 1\}$ represents entropies, and $\omega : \{1, 2, \ldots, m + 1\}$ is the probabilities of segmented classes $C : \{0, 1, \ldots, m\}$, respectively.

2.2 Otsu's Between Class Variance

Otsu's method is a nonparametric and unsupervised automatic threshold selection method [12]. It is used to choose the optimum threshold values by maximizing between class variance of the segmented classes. If we have to select, a total of m number of thresholds t: $\{1, 2,\ldots, m\}$. Then, the threshold values sub-partition the image into $(m + 1)$ number of classes such as, C: $\{0, 1, 2,\ldots, m\}$ in turn of maximizing the below objective function,

$$J(t : \{1, 2, \ldots, m\}) = \sigma_0^2 + \sigma_1^2 + \ldots + \sigma_m^2 \tag{3}$$

where $\sigma : \{0, 1, 2, \ldots, m\}$ are the class variances. $\omega : \{0, 1, 2, \ldots, m\}$ are the probabilities of the respective classes. $\mu : \{0, 1, 2, \ldots, m\}$ represent the mean levels for the segmented classes- $C : \{0, 1, 2, \ldots, m\}$, respectively.

2.3 Minimum Cross-Entropy

Lee et al. [13] first applied minimum cross-entropy measure to image segmentation. Let us consider, I is the actual image; L is gray-levels number, and h is represented for the histogram of image. If there are n number of threshold levels such as —$t : \{0, 1, 2, \ldots, m - 1\}$, which divide an image into $n + 1$ classes such as —$C : \{0, 1, \ldots, m\}$. These classes include pixels with the gray-level values within the intervals $(0, t_0)$, $(t_0 + 1, t_1)\ldots$, $(t_{m\text{-}1} +1, l{-}1)$. The optimum threshold values say —$t : \{0, 1, 2, \ldots, m - 1\}$ could be selected by minimizing the cross-entropy function as defined below.

$$\begin{aligned} D(t_1, t_2, \ldots, t_m) = {} & \sum_{i=1}^{L} ih(i)\log(i) - \sum_{i=1}^{t_1} ih(i)\log(\mu_0) \\ & - \sum_{i=t_1+1}^{t_2} ih(i)\log(\mu_1) - \ldots. - \sum_{i=t_m+1}^{L} ih(i)\log(\mu_m) \end{aligned} \tag{4}$$

where $\mu : \{0, 1, .., m\}$ defines the mean intensity values for the classes $C : \{0, 1, 2, \ldots, m\}$.

The above Eq. (4) defines the cross-entropy of the optimum threshold values $t : \{0, 1, 2, \ldots, m - 1\}$ between the actual image and the threshold image. For a specific image, the first term is always a constant, so we can modify the objective function as,

$$\eta(t_1, t_2, ..., t_m) = -\sum_{i=1}^{t_1} ih(i)\log(\mu_0) \\ - \sum_{i=t_1+1}^{t_2} ih(i)\log(\mu_1) - \ldots - \sum_{i=t_m+1}^{L} ih(i)\log(\mu_m) \quad (5)$$

2.4 Masi Entropy

Consider an image I that contains a total of L gray levels. Then, we can represent histogram of the image as a function of probability distribution. Hence, we can represent the normalized histogram for a gray-level value i as, $h_i = {}^{n_i}\!/_{(l * m)}$. This equation defines; gray level i occurs n_i times in the complete image. Then, we can represent $\sum_i h_i = 1$. Then, the Masi's entropy [15, 16] for bi-level thresholding can be represented as,

$$S_r\left({}^{I}\!/_{t}\right) = S_r(C_1) + S_r(C_2) \quad (6)$$

where t defines optimum threshold value found for the Masi's entropy in Eq. (6), which sub-divides the image into two classes as $C1$, $C2$ having class probabilities as ω_0, ω_1. These class probabilities can be formulated as, $\omega_0 = \sum_{i=0}^{t-1} h_i$ and $\omega_1 = \sum_{i=t}^{L-1} h_i$.

2.5 Tsalli's Entropy

Tsai et al. [17] had presented Tsalli's entropy, which is used as objective function (maximization) for multilevel thresholding concept for image segmentation [6, 18, 19]

2.6 Renyi's Entropy

Renyi et al. [20] proposed a new entropy concept called Renyi's entropy, which has been used as the objective function of multilevel-based image segmentation techniques [18, 21, 22].

3 Multilevel Thresholding Methods

Multilevel thresholding is a vast and widely explored research area under image processing. The basic method of multilevel thresholding can be roughly categorized into two sections as follows:

- **Parametric thresholding method**

The parametric methods are based on the histogram analysis, which are the basis of some mathematical functions. Farnoosh et al. [14] used Gaussian mixture method of image segmentation. They have used expectation maximization procedure for the purpose of parameter estimation. If x is random variable, now for a probability determination, the Gaussian mixture model can be derived as

$$f(x) = \sum_{i=1}^{k} P_i N\left(\frac{x}{\mu_i \sigma_i^2}\right) \tag{7}$$

where $k =$ number of regions. Huang et al. [23] had proposed an image thresholding technique based on Gaussian mixture method. In this, they had used the concept of expectation maximization for histogram analysis of an image. In this proposed method, an optimal threshold level is chosen by taking the average of Gaussian mixture means.

- **Nonparametric thresholding method**

In case of nonparametric approach, optimal thresholds are chosen based on the information of the classes. Otsu's and Kapur's entropic concept of thresholding are some most widely used nonparametric-based multi-thresholding techniques. Multilevel thresholding techniques are very much popular approaches. But, if we increase the total number of threshold, the computational complexity increases. Many algorithms had proposed to reduce this complexity of multilevel thresholding. These algorithms are broadly categorized into two types—iterative algorithm and meta-heuristic algorithm-based multilevel thresholding.

3.1 Iterative Multilevel Thresholding Techniques

Iterative multilevel thresholding techniques are helpful in reducing the computational complexity of multilevel thresholding-based image segmentation. There are various iterative multilevel thresholding techniques for image segmentation have been developed. Wang et al. [24] presented an iterative thresholding concept for multiphase segmentation. In their work, they had adopted the idea of Chan-vase model for multiphase image segmentation. They claimed that their method gives an optimal time complexity of $O(n \log n)$ per iteration, and this method is stable and has the property of total energy decay. Liao et al. [25] proposed a modified form

Otsu's method for multilevel thresholding. This modified function is pre-computed which reduces computational cost. The new objective function yields the similar levels of threshold values as of original Otsu's concept. This technique helpful to maintain an optimum time complexity. Shang et al. [26] had presented an iterative gradient-based multilevel thresholding concept, which is different from heuristic-based algorithm for multilevel thresholding. Yin et al. [27] had presented an iterative concept for multilevel thresholding technique and uses the concept of Otsu's and Kapur's entropic functions. In their methods, it begins with a bi-level thresholding, and in turn, they use the initial result to achieve the higher-level thresholds. This algorithm could determine the number of thresholds automatically. Reddi et al. [28] had suggested an iterative-based approach of multiple thresholds scheme for image segmentation. They used criterion function for the purpose of searching optimum threshold values. This method uses Otsu's function with an iterative procedure to segment an image. They consider image histogram as a continuous function for probability distribution. Arora et al. [29] had presented an iterative method in image segmentation for the concept of multilevel thresholding. This procedure selects the first, and the last peaks of the histogram and continuously apply this concept up to when an acceptable level of progress has been encountered in the segmentation.

3.2 Meta-Heuristic-Based Multilevel Thresholding Techniques

Meta-heuristic-based multilevel thresholding techniques are beneficial when we choose more numbers threshold values. With the use of meta-heuristic-based algorithms, we can maintain an optimum time complexity. Yin et al. [30] presented a genetic algorithm in image segmentation concept using multilevel thresholding method. It uses Kapur's and Otsu's concept as the objective function. This procedure claimed that it incurred a comparatively computational cost. Bhandari et al. [31] presented a concept of multilevel thresholding for image segmentation technique with the use of a beta differential evolution algorithm. Liu et al. [32] presented a concept of multilevel thresholding for image segmentation for breast cancer detection using a modified differential evolution algorithm. This method uses two basic concepts such as—two-dimensional histogram and Kapur's entropy. Mlakar el al. [33] presented a concept of multilevel thresholding for image segmentation technique using a hybrid concept DE algorithm. It uses Otsu's class variance concept to select optimum thresholds. Time complexity for this method is quite less (~0.5 s). Khalek et al. [18] had proposed an idea of image segmentation for brain MR images using multilevel thresholding concept. This method uses a two-dimensional concept of Tsallis and Renyi entropic function. Abualigah et al. [34] presented a concept of image segmentation technique based on multilevel thresholding for detecting COVID-19 CT images using a novel evolutionary optimization algorithm.

Khairuzzam et al. [35] presented a concept of image segmentation technique by multilevel thresholding using the concept of particle swarm optimization. They use the concept of Masi entropic function to select optimum threshold values. They used mean structural similarity and peak signal-to-noise ratio to evaluate the quality of segmented images. Gao et al. [36] presented a concept of image segmentation based on multilevel thresholding using a quantum nature particle swarm optimization algorithm. It uses Otsu's class variance concept for objective function to select optimum threshold values. Computational complexity of this method is moderate. Wang et al. [1] presented a concept of image segmentation technique based on multilevel thresholding using a hybrid quantum nature particle swarm optimization algorithm. It uses Kapur's entropic as function to select optimum thresholds.

Tang et al. [37] had presented a concept of image segmentation technique based on multilevel thresholding using modified bacterial foraging optimization. It uses Tsalli's entropic function to choose the optimum thresholds. Chouhan et al. [38] proposed an artificial neural network-based bacterial foraging optimization algorithm to segment plants leaf images. Khairuzzaman et al. [39] had presented a concept of image segmentation technique based on multilevel thresholding using gray wolf optimizer. It is a heuristic approach optimization algorithm; the basic concept of GWO is adopted based on the concept of societal and hunting characteristics of the wolfs. It uses Otsu's class variance function and Kapur's entropic function. Houssain et al. [40] had proposed a meta-heuristic-based black widow Optimization (BWO) algorithm for multilevel thresholding. It uses Otsu's class variance function and Kapur's entropic function to choose the best threshold values. The basic procedure of black widow optimization algorithm is based on the idea of natural evolution such as—selection, reproduction, and mutation. Black widow optimization algorithm follows fast convergence and eliminates the local optima problem. This algorithm guarantees the balance between exploration as well as exploitation. Aziz et al. [41] had presented a multilevel thresholding-based image segmentation using a hybrid concept of optimization algorithm. It uses moth–flame optimization (MFO) as well as whale optimization algorithm. Anitha et al. [42] presented a concept of image segmentation technique based on multilevel thresholding for color using modified whale optimization algorithm. It uses Otsu's function and Kapur's entropic function for their algorithm. Wang et al. [21] presented a concept of image segmentation technique based on multilevel thresholding for color image using Salp swarm optimization algorithm. They uses Kapur's entropic function, Reny's entropic function, and Otsu's functions to select optimum threshold values.

Bhandari et al. [43] had presented a concept of image segmentation technique based on multilevel thresholding using cuckoo search algorithm. They have considered Kapur's entropic function. This procedure is used to segment satellite images. Oliva et al. [7] presented a concept of image segmentation technique based on multilevel thresholding for MR images using crow search algorithm. This method uses minimum cross-entropic functions. Cuevas et al. [44] presented a multilevel thresholding approach for image segmentation using artificial bee colony algorithm. Yue et al. [45] presented a concept of image segmentation technique based on multilevel thresholding using a hybrid version of bat optimization algorithm. Xu et al. [46]

presented a concept of image segmentation technique based on multilevel thresholding for color image using a modified dragonfly algorithm. They used minimum cross-entropy and Otsu's class variance functions. Sharma et al. [6] had suggested a concept of image segmentation technique based on multilevel thresholding using firefly algorithm. It uses Kapur's and Tsalli's entropic function to choose optimum threshold values. Shingh et al. [47] presented a multilevel thresholding-based image segmentation technique using a hybrid concept of dragonfly and firefly algorithm. Their method uses Kapur's entropic function and Otsu's function for threshold selection.

Sarkar et al. [22] had proposed a multilevel thresholding-based segmentation technique for natural as well as medical images. This method uses Renyi's and cross-entropic functions for threshold selection. Rajinikanth et al. [47] had presented a segmentation technique for MR images. This method uses a hybrid concept of teaching learning optimization algorithm. Upadhyay et al. [48] had presented a multilevel thresholding-based image segmentation technique using a hybrid concept of meta-heuristic algorithms as follows: particle swarm optimization and artificial bee colony optimization algorithms. It uses Kapur's entropic function to select optimum threshold values.

4 Performance Measure Metrics and Relevant Datasets

Some widely used dataset to evaluate the multilevel thresholding-based image segmentation techniques are USC-SIPI image dataset, BSD 500, Berkeley database for color image, Berkeley segmentation dataset 500 (BSDS 500), University of Wisconsin dataset, etc., and some common performance measure indicator to evaluate the image segmentation techniques using multilevel thresholding concept are mean structural similarity index (MSSIM), peak signal-to-noise ratio (PSNR), structural similarity index (SSIM), feature similarity index (FSIM), mean squared error (MSE), Wilcoxon Rank-sum test.

- **Mean structural similarity index**

This measure is used to check the quality of segmented images, and it is represented as,

$$\mathrm{MSSIM}(A, B) = \frac{\sum_{j=1}^{M} \mathrm{SSIM}\left(x_j y_j\right)}{M} \tag{8}$$

where A and B define the original and the segmented image. x_i and y_j denote the image data in jth local window. M is the total number of window.

- **Peak signal-to-noise ratio**

It is the performance measure calculator used to measure the similarity of the segmented image against actual image and is represented as,

$$\text{PSNR} = 10\log_{10}\left(\frac{255^2}{\text{MSE}}\right) \tag{9}$$

Here, MSE defines the mean squared error, and it is represented as,

$$\text{MSE} = \frac{1}{\text{MN}}\sum_{p=1}^{M}\sum_{q=1}^{N}[O(p,q) - S(p,q)] \tag{10}$$

where $O(p, q)$ and $S(p, q)$ are the gray level of original and the segmented images in the pth and qth column, respectively.

- **Structural similarity index**

This similarity index is used to calculate the similarity of an actual image and the segmented image; it is represented as,

$$\text{SSIM}_{(i,j)} = \frac{(2\mu_i\mu_j + C_1)(2\sigma_{ij} + C_2)}{\left(\mu_i^2 + \mu_j^2 + C_1\right)\left(\sigma_i^2 + \sigma_j^2 + C_2\right)} \tag{11}$$

where μ_i and μ_j are the mean intensities, ${\sigma_i}^2$ and ${\sigma_j}^2$ represent the standard deviation of the actual image and the extracted image. σ_{ij} represents the covariance of the actual and the extracted image.

- **Feature similarity index**

This index is used to measure the quality of an image by evaluating the feature similarity of the actual image and the segmented image, and it is represented as,

$$\text{FSIM} = \frac{\sum_{i\in\Omega} S_L(i) \times PC_m(i)}{\sum_{i\in\Omega} PC_m(i)} \tag{12}$$

where Ω defines the number of pixel, $S_L(x)$ represents the similarity score, and $\text{PC}_m(x)$ represents the phase consistency measure.

The performance measure metrics, relevant data sets, and respective objective functions used in various multilevel thresholding-based image segmentation techniques using some recent meta-heuristic algorithms have been presented in Table 1.

Table 1 Comparison of some recent multilevel thresholding algorithms

Paper refs	Objective function	Proposed algorithm	Performance metrics	Data set
Khairuzzaman and Chaudhury [35]	Masi entropic function	Particle swarm optimization	MSSIM, PSNR	USC-SIPI image
Houssain et al. [40]	Otsu's between class variance, Kapur's entropy	Black widow optimization (BWO)	SSIM, PSNR, FSIM	
Khairuzzaman and Chaudhury [39]	Otsu's between class variance, Kapur's entropy	Gray wolf Optimizer (GWO)	MSSIM	USC-SIPI image dataset, BSD 500
Xu et al. [46]	Otsu's function, minimum cross-entropy	Modified dragonfly algorithm	Standard deviation, average fitness values, PSNR, SSIM, FSIM	Berkeley database for color image
Wang et al. [21]	Otsu's function, Kapur's entropy, Renyi's entropy	Salp swarm optimization algorithm	PSNR, best fitness values, FSIM	Berkeley image database for color image
Sharma et al. [6]	Kapur's entropy, Tsalli's entropy	Firefly optimization algorithm	PSNR, SSIM	Berkeley image dataset
Singh et al. [47]	Otsu's function, Kapur's entropy	Hybrid version of dragonfly and firefly algorithm	PSNR, mean, threshold values, number of iterations taken to converge	BSDS 500
Anitha et al. [42]	Otsu's function, Kapur's entropy	Modified whale optimization algorithm	PSNR, SSIM, FSIM, MSE, CPU computing time, Wilcoxon test	University of Wisconsin BSD 500, Landsat Image Gallery (NASA)
Wang et al. [20]	2D-Kapur's entropy	Hybrid quantum behaved PSO	High segmentation accuracy, good convergence, anti-noise	USC-SIPI image database
Yan et al. [49]	Kapur's entropy	Whale optimization algorithm	PSNR, SSIM, Wilcoxon Rank-sum test	Own image dataset

5 Conclusion and Future Scope

In this study, we discussed about various multilevel thresholding techniques used for image segmentation. We summarized some of the important objective functions as follows—Otsu's class variance function, Kapur's entropic function, minimum cross-entropy, Masi entropy, Renyi's and Tsalli's entropic functions, etc., used in automatic multilevel thresholding for the purpose of choosing optimum thresholds. In multilevel thresholding techniques computational complexity and to maintain the quality of the images is still a challenging issue. Several meta-heuristic algorithms are being used in this regard such as—particle swarm optimization, whale optimization algorithm, gray wolf optimizer, black widow optimization algorithm, Salp swarm optimization, dragonfly algorithm, firefly optimization algorithm, ant-colony optimization, and bacterial foraging algorithm.

Performances of multilevel thresholding-based image segmentation can be increased by tuning a better combination of objective function and the meta-heuristic algorithms. Modified objective functions are being used widely. Hybrid version or modified or some time new meta-heuristic algorithms are used on a large scale to increase the performances. To evaluate the performances, the most widely used performance metrics are—MSSIM, PSNR, SSIM, FSIM, MSE, CPU computing time, Wilcoxon test, etc.

In the future scope, a modified and hybrid concept of the objective functions can be used for multilevel thresholding technique. Further, it can be used some of the recent meta-heuristic algorithms or a hybrid concept of new meta-heuristic algorithm and objective functions for automatic multilevel thresholding-based image segmentation techniques.

Acknowledgements This work is supported by Science and Engineering Research Board (SERB), Department of Science and Technology (DST), Government of India under Grant No. EEQ/2019/000657.

References

1. Choy SK, Lam SY, Yu KW, Lee WY, Leung KT (2017) Fuzzy model-based clustering and its application in image segmentation. Pattern Recogn 68:141–157
2. Jung C, Jian M, Liu J, Jiao L, Shen Y (2014) Interactive image segmentation via kernel propagation. Pattern Recogn 47(8):2745–2755
3. Rodríguez-Esparza E, Zanella-Calzada LA, Oliva D, Pérez-Cisneros M (2020) Automatic detection and classification of abnormal tissues on digital mammograms based on a bag-of-visual-words approach. In: Medical Imaging 2020: Computer-Aided Diagnosis (vol 11314). International Society for Optics and Photonics, p. 1131424
4. Montalvo M, Guijarro M, Ribeiro A (2018) A novel threshold to identify plant textures in agricultural images by Otsu and Principal Component Analysis. J Intell Fuzzy Syst 34(6):4103–4111
5. Sengar SS, Mukhopadhyay S (2019) Motion segmentation-based surveillance video compression using adaptive particle swarm optimization. Neural Comp Appl, pp 1–15
6. Sharma A, Chaturvedi R, Kumar S, Dwivedi UK (2020) Multi-level image thresholding based on Kapur and Tsallis entropy using firefly algorithm. J Interdis Math 23(2):563–571
7. Oliva D, Hinojosa S, Cuevas E, Pajares G, Avalos O, Galvez J (2017) Cross entropy based thresholding for magnetic resonance brain images using crow search algorithm. Expert Syst Appl 79:164–180

8. Guo Y, Ashour AS (2019) Neutrosophic sets in dermoscopic medical image segmentation. In: Neutrosophic set in medical image analysis. Academic Press, pp 229–243
9. Raju PDR, Neelima G (2012) Image segmentation by using histogram thresholding. Int J Comp Sci Eng Tech 2(1):776–779
10. Tsai WH (1985) Moment-preserving thresolding: a new approach. Comput Vis Graph Image Process 29(3):377–393
11. Kapur JN, Sahoo PK, Wong AK (1985) A new method for gray-level picture thresholding using the entropy of the histogram. Computer Vision, Graph Image Process 29(3):273–285
12. Otsu N (1979) A threshold selection method from gray-level histograms. IEEE Trans Syst Man Cybern 9(1):62–66
13. Li CH, Lee CK (1993) Minimum cross entropy thresholding. Pattern Recogn 26(4):617–625
14. Farnoush R, Zar PB, Image segmentation using Gaussian mixture model.
15. Masi M (2005) A step beyond Tsallis and Rényi entropies. Phys Lett A 338(3):217–224
16. Nie F, Zhang P, Li J, Ding D (2017) A novel generalized entropy and its application in image thresholding. Signal Process 134:23–34
17. Rényi A (1961) On measures of entropy and information. In Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics, January, pp 547–561. University of California Press
18. Sarkar S, Das S, Chaudhuri SS (2017) Multi-level thresholding with a decomposition-based multi-objective evolutionary algorithm for segmenting natural and medical images. Appl Soft Comput 50:142–157
19. Jena B, Naik MK, Panda R, Abraham A (2021) Maximum 3D Tsallis entropy based multilevel thresholding of brain MR image using attacking Manta Ray foraging optimization. Eng Appl Artif Intell 103:104293
20. Wang HQ, Cheng XW, Chen GC (2021) A hybrid adaptive quantum behaved particle swarm optimization algorithm based multilevel thresholding for image segmentation. In: 2021 IEEE International Conference on Information Communication and Software Engineering (ICICSE), pp 97–102. IEEE
21. Wang S, Jia H, Peng X (2020) Modified salp swarm algorithm based multilevel thresholding for color image segmentation. Math Biosci Eng 17(1):700–724
22. Rajinikanth V, Satapathy SC, Fernandes SL, Nachiappan S (2017) Entropy based segmentation of tumor from brain mr images–a study with teaching learning based optimization. Pattern Recogn Lett 94:87–95
23. Huang Z-K, Chau K-W (2008) A new image thresholding method based on gaussian mixture model. Appl Math Comput 205(2):899–907
24. Wang D, Li H, Wei X, Wang X-P (2017) An efficient iterative thresholding method for image segmentation. J Comput Phys 350:657–667
25. Liao P-S, Chen T-S, Chung P-C et al (2001) A fast algorithm for multilevel thresholding. J Inf Sci Eng 17(5):713–727
26. Shang C, Zhang D, Yang Y (2021) A gradient-based method for multilevel thresholding. Expert Syst Appl 175:114845
27. Yin P-Y, Chen L-H (1997) A fast iterative scheme for multilevel thresholding methods. Signal Process 60(3):305–313
28. Reddi S, Rudin S, Keshavan H (1984) An optimal multiple threshold scheme for image segmentation. IEEE Trans Syst Man Cybern 4:661–665
29. Arora S, Acharya J, Verma A, Panigrahi PK (2008) Multilevel thresholding for image segmentation through a fast statistical recursive algorithm. Pattern Recogn Lett 29(2):119–125
30. Yin P-Y (1999) A fast scheme for optimal thresholding using genetic algorithms. Signal Process 72(2):85–95
31. Bhandari AK (2020) A novel beta differential evolution algorithm-based fast multilevel thresholding for color image segmentation. Neural Comput Appl 32(9):4583–4613
32. Liu L, Zhao D, Yu F, Heidari AA, Ru J, Chen H, Pan Z (2021) Performance optimization of differential evolution with slime mould algorithm for multilevel breast cancer image segmentation. Comput Biol Med 138:104910

33. Mlakar U, Potocnik B, Brest J (2016) A hybrid differential evolution for optimal multilevel image thresholding. Expert Syst Appl 65:221–232
34. Abualigah L, Diabat A, Sumari P, Gandomi AH (2021) A novel evolutionary arithmetic optimization algorithm for multilevel thresholding segmentation of covid-19 CT images. Processes 9(7):1155
35. Khairuzzaman AKM, Chaudhury S (2019) Masi entropy based multilevel thresholding for image segmentation. Multimedia Tools Appl 78(23):33573–33591
36. Gao H, Xu W, Sun J, Tang Y (2009) Multilevel thresholding for image segmentation through an improved quantum-behaved particle swarm algorithm. IEEE Trans Instrum Meas 59(4):934–946
37. Tang K, Xiao X, Wu J, Yang J, Luo L (2017) An improved multilevel thresholding approach based modified bacterial foraging optimization. Appl Intell 46(1):214–226
38. Chouhan SS, Kaul A, Sinzlr UP (2019) Plants leaf segmentation using bacterial foraging optimization algorithm. In: 2019 International Conference on Communication and Electronics Systems (ICCES), July, pp 1500–1505. IEEE
39. Khairuzzaman AKM, Chaudhury S (2017) Multilevel thresholding using grey wolf optimizer for image segmentation. Expert Syst Appl 86:64–76
40. Houssein EH, Helmy BE-D, Oliva D, Elngar AA, Shaban H (2021) A novel black widow optimization algorithm for multilevel thresholding image segmentation. Expert Syst Appl 167:114159
41. Abdel AM, Ewees AA, Hassanien AE (2017) Whale optimization algorithm and moth-flame optimization for multilevel thresholding image segmentation. Expert Syst Appl 83:242–256
42. Anitha J, Pandian SIA, Agnes SA (2021) An efficient multilevel color image thresholding based on modified whale optimization algorithm. Expert Syst Appl 178:115003
43. Bhandari AK, Singh VK, Kumar A, Sing GK (2014) Cuckoo search algorithm and wind driven optimization based study of satellite image segmentation for multilevel thresholding using kapur's entropy. Expert Syst Appl 41(7):3538–3560
44. Cuevas E, Sencion F, Zaldivar D, Perez-Cisneros M, Sossa H (2012) A multi-threshold segmentation approach based on artificial bee colony optimization. Appl Intell 37(3):321–336
45. Yue X, Zhang H (2020) Modified hybrid bat algorithm with genetic crossover operation and smart inertia weight for multilevel image segmentation. Appl Soft Comput 90:106157
46. Xu L, Jia H, Lang C, Peng X, Sun K (2019) A novel method for multilevel color image segmentation based on dragonfly algorithm and differential evolution. IEEE Access 7:19502–19538
47. Singh S, Mittal N, Singh H (2021) A multilevel thresholding algorithm using HDAFA for image segmentation. Soft Comput 25(16):10677–10708
48. Upadhyay P, Chhabra JK (2021) Multilevel thresholding based image segmentation using new multistage hybrid optimization algorithm. J Ambient Intell Humaniz Comput 12:1081–1098
49. Yan Z, Zhang J, Yang Z, Tang J (2020) Kapur's entropy for underwater multilevel thresholding image segmentation based on whale optimization algorithm. IEEE Access 9:41294–41319
50. Abdel-Khalek S, Ishak AB, Omer OA, Obada A-S (2017) A two-dimensional image segmentation method based on genetic algorithm and entropy. Optik 131:414–422

A Novel Convolutional Neural Network-Based Segmentation Model for Lung CT Scan Images Affected by COVID-19

Varun Srivastava, Nikhil Kalra, Ayushi Tulsyan, and Romy Kumari

Abstract In recent times, the detection of COVID by lung CT scan images has become an active field of research due to the increase in the number of COVID cases worldwide. COVID causes lesion-based damage in the lungs which can be easily analyzed by a CT scan image. The proposed methodology uses a publicly available database of lung computed tomography (CT) scan images collected from 297 subjects having 8739 scans and thereby apply a Covi-Net model for lesion-based segmentation and thereby COVID detection. The Covi-Net model is an extension of U-Net model used for biomedical image classification. The model outperformed related algorithms with a dice value of 0.886.

Keywords COVID-19 detection · Biomedical image classification · Dice coefficient · COVID-19

1 Introduction

In December 2019, a novel respiratory tract disease called COVID-19 was found which was caused due to a virus called SARS-COV-2. Since the virus was able to spread very swiftly by human to human transmission, the situation has been declared a pandemic in January 2020 [1].

Till now, reverse-transcription-polymerase chain reaction (RT-PCR) is one of the most common diagnostic methods for detecting nucleotides from materials obtained by oropharyngeal swab, nasopharyngeal swab, bronchoalveolar lavage, or tracheal aspirate [2, 3]. However, recent findings have indicated that RT-PCR's sensitivity in the detection of COVID-19 has been poor [4, 5]. This could be due to specimen quality or lack of viral material in the sample being tested. COVID-19 patients' chest's scan

V. Srivastava (✉)
Thapar Institute of Engineering and Technology, Patiala, Punjab, India
e-mail: varun.srivastava@thapar.edu

N. Kalra · A. Tulsyan · R. Kumari
Bharati Vidyapeeth's College of Engineering, New Delhi, India

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_60

images, on the other hand, frequently display bilateral patchy shadows or ground glass opacity in the lung [6], hence CT has become an important component of the diagnostic process.

However, with increase in confirmed and probable COVID-19 cases, the time taking process of physically contouring lung lesions has become obsolete. Building a quick automated segmentation for COVID-19 infection is critical for illness evaluation to speed up diagnosis and improve access to therapy.

Wang et al. [15] created an algorithm for extracting COVID-19's graphical features and providing a clinical diagnosis before the pathogenic test.

To discover COVID-19, Ayrton [16] used the transfer learning technique using the ResNet50 backbone. Wang et al. [14] proposed a DCNN that was built to recognize COVID-19 patients from tomography images. Multiple deep learning approaches have been explained in [17] for the prognosis of COVID-19. Gozes et al. [18] introduced a system that combines 2D and 3D deep-NN models with clinical knowledge. They modified and extended existing deep network models.

Tang et al. [19] quantify severity by using quantitative analysis through random forest algorithm. For classification, Shi et al. [20] proposed SA random forest technique to combat infection (iSARF). In [21], authors created a method based on deep learning for segmenting and quantifying infection areas from chest images. Deep learning models have clearly shown a major role in diagnosis of diseases in biomedical CT scans. Some of the techniques are given in [7–13].

Further, many deep learning pedagogies have been proposed by various researchers for the prediction and segmentation of COVID-19 in lung biomedical images. In [22], authors trained a ResNet model for the classification of X-Ray images of COVID patients taken from multiple sources. The authors claimed accuracy of around 98%.

In [23], nine architectures, viz. Baseline CNN, DenseNet201, VGG16, VGG19, Inception_ResNet_V2, Inception V3, Xception, ResNet50, and MobileNet_V2 have been used for the classification of X-Rays with Pnemonia and COVID-19. All the architectures have been compared and MobileNet_V2, Inception_Resnet_V2, and ResNet50 were found to be performing better than 96%. Authors in [24] used deep learning to distinguish Coronavirus from simple influenza (a viral fever/pneumonia). A live dataset has been used from three hospitals in Zhejiang province of China with 618 sample images. The architecture achieved an accuracy of around 86.7%.

Authors in [25] introduced a COVIDX-Net model for diagnosing COVID-19 using X-Ray images. The classifier has been built on seven deep learning models, viz. DenseNet121, Xception, Inception V3, Inception-ResNet-V2, ResNet-V2, etc. Out of all these models, VGG19 and DenseNet outperformed with 90 f1-score in average for both classes—healthy and COVID—affected.

In [26], ResNet50, Inception-ResNet-V2, and Inception V3 model have been implemented for a dataset collected from 50 COVID patients and 50 normal ones. ResNet outperformed all other models in terms of accuracy for COVID detection. Authors in [27] have used 300 CT images to first segment the image for COVID-affected area and then cropped the affected area. Four subsets of patches have been created and features like gray-level cooccurence matrix, gray-level size zone matrix, local directional pattern, last discrete wavelet transform, etc., have been extracted. An accuracy of around 99.68% has been achieved through this.

In [28], authors used deep learning models like VGG16 and VGG19. on images from GitHub repository with images from 75 subjects (50 COVID-19 and 25 healthy) have been used for segmentation followed by classification of images using support vector machine (SVM) model. The model with SVM and ResNet50 performed the best with 95.52% accuracy.

Few more techniques for segmentation and detection of COVID-19 in patients are summarized in [29–34].

In this research, we attempt to develop a new customized deep COVID-19-infected chest CT image detection model. The images are first segmented using a convolutional neural network (CNN). Groundglass opacities (GGOs), areas of consolidation, and a combination of both can be seen in all lung lobes in the chest CT images with COVID-19 infection as shown in Fig. 1. The most of lesions were observed in the periphery, with a modest preference for dorsal lung locations.

The boundaries of a lung CT scan image are hard to distinguish from the chest wall, making their segmentation difficult. Annotation processing involves adjusting various characteristics such as window width and window locations. As shown in Fig. 1, this adjustment reveals the limits of COVID-19 infection zones, which further leads to COVID-19 infection picture segmentation.

The primary contributions of the proposed research work are summarized as follows:

- For the segmentation infected areas from CT images, a unique DL network (Covi-Net) is proposed. Covi-Net is a three-dimensional (3D) convolutional DL system that uses chest images to autonomously segment COVID-19 infected spots and the overall lung. To compute the dice coefficient (DC), a model is suggested to segment the infection mask from the chest image and thereby compute various loss functions followed by boundary surface loss function.

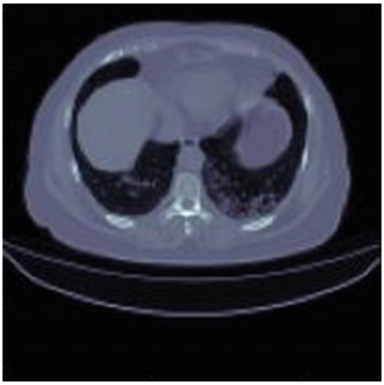


Fig. 1 Chest CT image of a patient suffering with COVID-19

- The problem of boundary loss during segmentation is addressed, which is a major issue in the detection of COVID-19 infection regions, is addressed using a surface loss function.

The paper has been arranged in the following manner. Section 2 describes the methodology and step-by-step algorithm for the segmentation of the COVID-19-affected region. Section 3 summarizes the results and establishes the superiority of the proposed work over related methodologies. Section 4 presents the conclusion and future scope.

2 Methodology

A. Dataset

A publicly available dataset from [35] has been used which contains 349 COVID-19 CT images from 216 patients and 463 non-COVID-19 CTs.

B. Proposed Algorithm

(1) Preprocessing

The input image is first normalized. The pixel intensities that do not have major information are cropped. This is done by using Hounsfield Units (HU) transform. Hounsfield transform computes the CT units in an interpretable format for easy extraction of relevant information from an image. HU transform is a scale to measure the radiodensity of CT scan images. In CT scan images, the HU value is also termed as CT number.

Thereby the pixels having maximum information can be plotted against Housfield Units. Figure 2 shows the various pixel intensity values in CT images. The intensity range from -400 to 600 had the maximum information, and therefore, all other intensity values are trimmed.

(2) Architecture

Covi-Net is a semantic segmentation architecture used for the segmentation of the COVID-19 imges. It is a fine-tuned U-Net model used for classification. In the traditional U-Net model, more convolutional and pooling layers are added with stride as 2 and filter size as 2×2 to obtain the Covi-Net model.

Figure 3 presents the Covi-Net model with layers and output at each layer used in the proposed methodology. It has two 3×3 convolutions separated by a 2×2 max-pooling layer with stride-value as 2 and a corrected linear unit (ReLU). The number of feature channels was raised by four times with each down-sampling step. A 2×2 convolution layer reduces feature channel to half, followed by a merger with feature map from diminishing path. Then two layers with kernel size 3×3, each having ReLU in expansive path are used for further segmentation. Total parameters used have been 35,238,293.

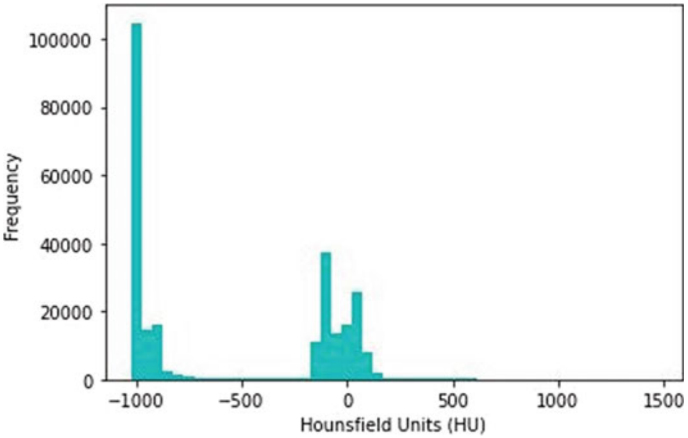


Fig. 2 Hounsfield units (HU) of the input CT scan image of lungs with COVID-19

To avoid border pixel loss that occur after every layer, cropping is essential. The last layer uses 1 × 1 convolution to convert each 64-component maps to the number of output classes. The network comprises a total of 23 convolutional layers.

(3) **Computation of Loss Functions**

The following loss functions were used:

1. Dice Loss: The Term Dice Loss Comes from the Srensen–Dice Coefficient, Which is Used to Compare Two Samples as Given in Eq. (1).

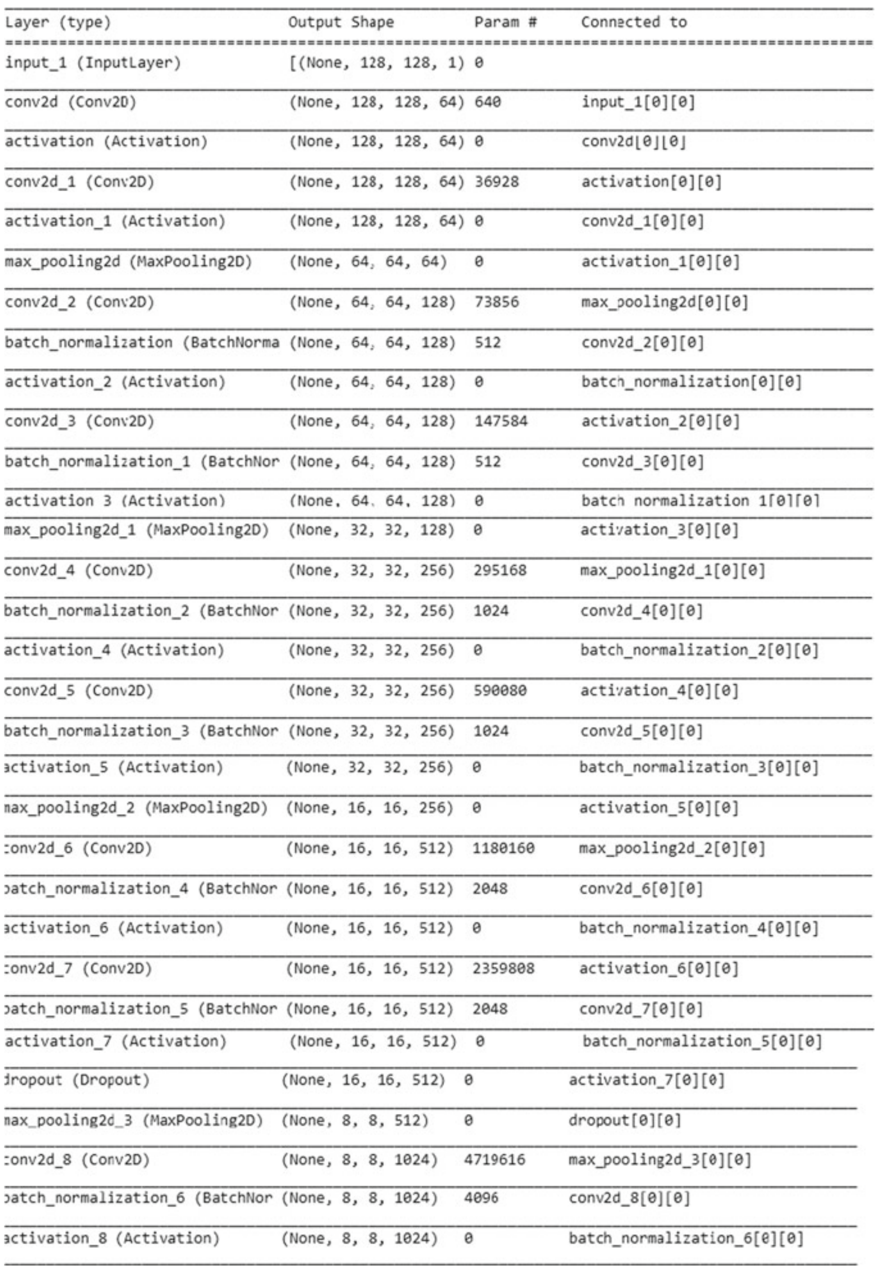
$$\mathrm{DC} = \frac{2\sum_{i}^{n} p_i * g_i}{\sum_{i}^{n} p_i^2 * \sum_{i}^{n} g_i^2} \tag{1}$$

Here p_i and g_i are pixel pairs with predicted and ground truth values.

(4) **Segmentation**

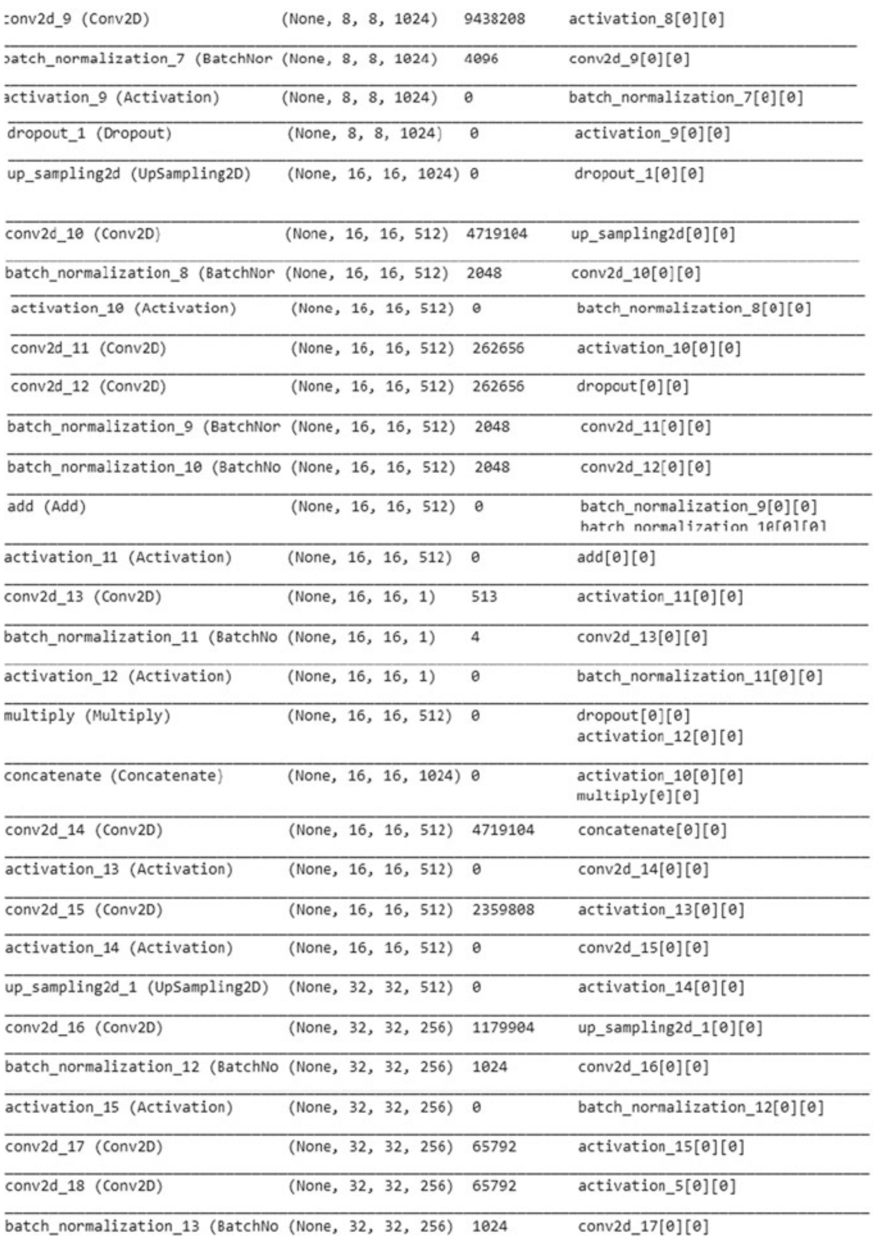
On the test set, the Covi-Net’s performance evaluation has been done for the lung segmentation task. After segmentation, the findings are quite near to being manually annotated. On comparison with similar approaches, the U-Net+ + approach frequently misses the lung’s boundary. The VNet approach is unable to provide a smooth lung segmentation boundary, but the proposed model pass the test without any difficulties.

As demonstrated in Fig. 4, the intensity of COVID-19 infection areas is remarkably similar to that of the lung. The areas damaged due to COVID-19 have been highlighted with a purple marker. Also Fully CN, U-Net, and Virtual-Net results in boundary



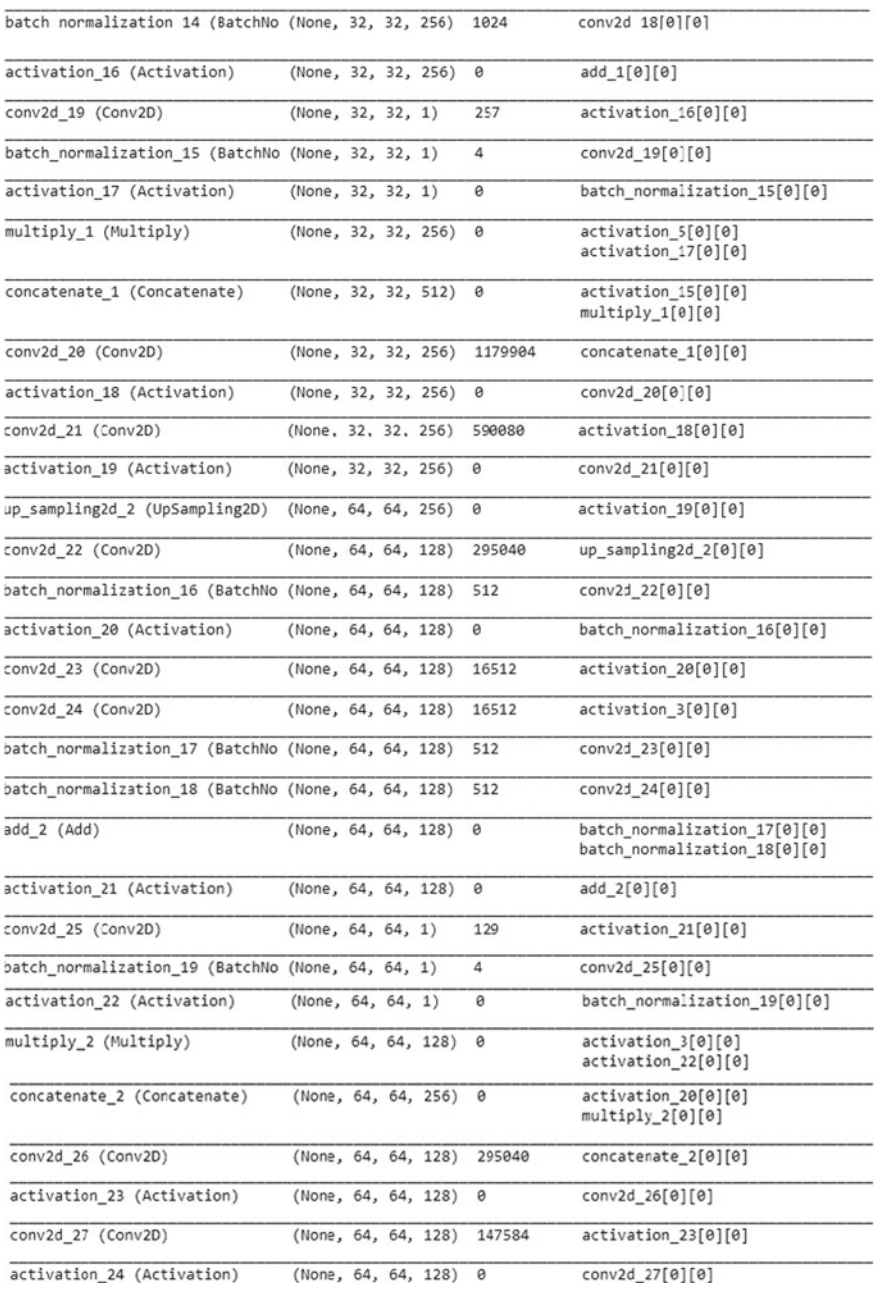
```
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_1 (InputLayer)            [(None, 128, 128, 1) 0
__________________________________________________________________________________________________
conv2d (Conv2D)                 (None, 128, 128, 64) 640         input_1[0][0]
__________________________________________________________________________________________________
activation (Activation)         (None, 128, 128, 64) 0           conv2d[0][0]
__________________________________________________________________________________________________
conv2d_1 (Conv2D)               (None, 128, 128, 64) 36928       activation[0][0]
__________________________________________________________________________________________________
activation_1 (Activation)       (None, 128, 128, 64) 0           conv2d_1[0][0]
__________________________________________________________________________________________________
max_pooling2d (MaxPooling2D)    (None, 64, 64, 64)   0           activation_1[0][0]
__________________________________________________________________________________________________
conv2d_2 (Conv2D)               (None, 64, 64, 128)  73856       max_pooling2d[0][0]
__________________________________________________________________________________________________
batch_normalization (BatchNorma (None, 64, 64, 128)  512         conv2d_2[0][0]
__________________________________________________________________________________________________
activation_2 (Activation)       (None, 64, 64, 128)  0           batch_normalization[0][0]
__________________________________________________________________________________________________
conv2d_3 (Conv2D)               (None, 64, 64, 128)  147584      activation_2[0][0]
__________________________________________________________________________________________________
batch_normalization_1 (BatchNor (None, 64, 64, 128)  512         conv2d_3[0][0]
__________________________________________________________________________________________________
activation_3 (Activation)       (None, 64, 64, 128)  0           batch_normalization_1[0][0]
__________________________________________________________________________________________________
max_pooling2d_1 (MaxPooling2D)  (None, 32, 32, 128)  0           activation_3[0][0]
__________________________________________________________________________________________________
conv2d_4 (Conv2D)               (None, 32, 32, 256)  295168      max_pooling2d_1[0][0]
__________________________________________________________________________________________________
batch_normalization_2 (BatchNor (None, 32, 32, 256)  1024        conv2d_4[0][0]
__________________________________________________________________________________________________
activation_4 (Activation)       (None, 32, 32, 256)  0           batch_normalization_2[0][0]
__________________________________________________________________________________________________
conv2d_5 (Conv2D)               (None, 32, 32, 256)  590080      activation_4[0][0]
__________________________________________________________________________________________________
batch_normalization_3 (BatchNor (None, 32, 32, 256)  1024        conv2d_5[0][0]
__________________________________________________________________________________________________
activation_5 (Activation)       (None, 32, 32, 256)  0           batch_normalization_3[0][0]
__________________________________________________________________________________________________
max_pooling2d_2 (MaxPooling2D)  (None, 16, 16, 256)  0           activation_5[0][0]
__________________________________________________________________________________________________
conv2d_6 (Conv2D)               (None, 16, 16, 512)  1180160     max_pooling2d_2[0][0]
__________________________________________________________________________________________________
batch_normalization_4 (BatchNor (None, 16, 16, 512)  2048        conv2d_6[0][0]
__________________________________________________________________________________________________
activation_6 (Activation)       (None, 16, 16, 512)  0           batch_normalization_4[0][0]
__________________________________________________________________________________________________
conv2d_7 (Conv2D)               (None, 16, 16, 512)  2359808     activation_6[0][0]
__________________________________________________________________________________________________
batch_normalization_5 (BatchNor (None, 16, 16, 512)  2048        conv2d_7[0][0]
__________________________________________________________________________________________________
activation_7 (Activation)       (None, 16, 16, 512)  0           batch_normalization_5[0][0]
__________________________________________________________________________________________________
dropout (Dropout)               (None, 16, 16, 512)  0           activation_7[0][0]
__________________________________________________________________________________________________
max_pooling2d_3 (MaxPooling2D)  (None, 8, 8, 512)    0           dropout[0][0]
__________________________________________________________________________________________________
conv2d_8 (Conv2D)               (None, 8, 8, 1024)   4719616     max_pooling2d_3[0][0]
__________________________________________________________________________________________________
batch_normalization_6 (BatchNor (None, 8, 8, 1024)   4096        conv2d_8[0][0]
__________________________________________________________________________________________________
activation_8 (Activation)       (None, 8, 8, 1024)   0           batch_normalization_6[0][0]
__________________________________________________________________________________________________
```

Fig. 3 Various layers of the Covi-Net model, Output at each layer and the next layer connected to the previous layer



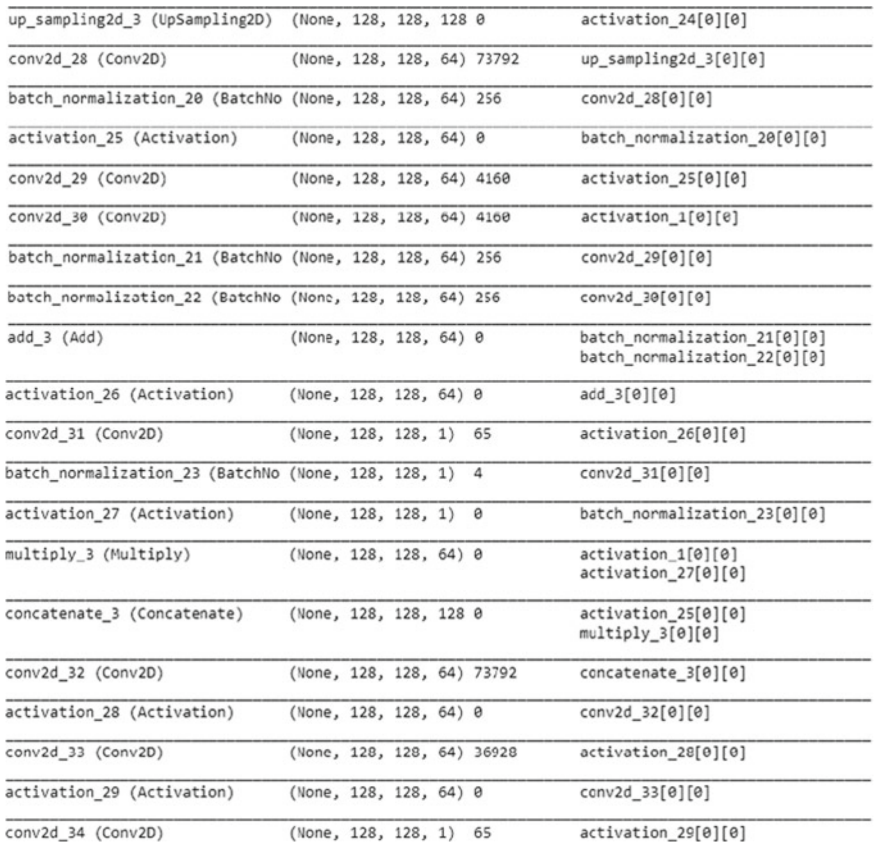
```
conv2d_9 (Conv2D)               (None, 8, 8, 1024)   9438208     activation_8[0][0]
________________________________________________________________________________________
batch_normalization_7 (BatchNor (None, 8, 8, 1024)   4096        conv2d_9[0][0]
________________________________________________________________________________________
activation_9 (Activation)       (None, 8, 8, 1024)   0           batch_normalization_7[0][0]
________________________________________________________________________________________
dropout_1 (Dropout)             (None, 8, 8, 1024)   0           activation_9[0][0]
________________________________________________________________________________________
up_sampling2d (UpSampling2D)    (None, 16, 16, 1024) 0           dropout_1[0][0]

________________________________________________________________________________________
conv2d_10 (Conv2D)              (None, 16, 16, 512)  4719104     up_sampling2d[0][0]
________________________________________________________________________________________
batch_normalization_8 (BatchNor (None, 16, 16, 512)  2048        conv2d_10[0][0]
________________________________________________________________________________________
activation_10 (Activation)      (None, 16, 16, 512)  0           batch_normalization_8[0][0]
________________________________________________________________________________________
conv2d_11 (Conv2D)              (None, 16, 16, 512)  262656      activation_10[0][0]
________________________________________________________________________________________
conv2d_12 (Conv2D)              (None, 16, 16, 512)  262656      dropout[0][0]
________________________________________________________________________________________
batch_normalization_9 (BatchNor (None, 16, 16, 512)  2048        conv2d_11[0][0]
________________________________________________________________________________________
batch_normalization_10 (BatchNo (None, 16, 16, 512)  2048        conv2d_12[0][0]
________________________________________________________________________________________
add (Add)                       (None, 16, 16, 512)  0           batch_normalization_9[0][0]
                                                                 batch_normalization_10[0][0]
________________________________________________________________________________________
activation_11 (Activation)      (None, 16, 16, 512)  0           add[0][0]
________________________________________________________________________________________
conv2d_13 (Conv2D)              (None, 16, 16, 1)    513         activation_11[0][0]
________________________________________________________________________________________
batch_normalization_11 (BatchNo (None, 16, 16, 1)    4           conv2d_13[0][0]
________________________________________________________________________________________
activation_12 (Activation)      (None, 16, 16, 1)    0           batch_normalization_11[0][0]
________________________________________________________________________________________
multiply (Multiply)             (None, 16, 16, 512)  0           dropout[0][0]
                                                                 activation_12[0][0]
________________________________________________________________________________________
concatenate (Concatenate)       (None, 16, 16, 1024) 0           activation_10[0][0]
                                                                 multiply[0][0]
________________________________________________________________________________________
conv2d_14 (Conv2D)              (None, 16, 16, 512)  4719104     concatenate[0][0]
________________________________________________________________________________________
activation_13 (Activation)      (None, 16, 16, 512)  0           conv2d_14[0][0]
________________________________________________________________________________________
conv2d_15 (Conv2D)              (None, 16, 16, 512)  2359808     activation_13[0][0]
________________________________________________________________________________________
activation_14 (Activation)      (None, 16, 16, 512)  0           conv2d_15[0][0]
________________________________________________________________________________________
up_sampling2d_1 (UpSampling2D)  (None, 32, 32, 512)  0           activation_14[0][0]
________________________________________________________________________________________
conv2d_16 (Conv2D)              (None, 32, 32, 256)  1179904     up_sampling2d_1[0][0]
________________________________________________________________________________________
batch_normalization_12 (BatchNo (None, 32, 32, 256)  1024        conv2d_16[0][0]
________________________________________________________________________________________
activation_15 (Activation)      (None, 32, 32, 256)  0           batch_normalization_12[0][0]
________________________________________________________________________________________
conv2d_17 (Conv2D)              (None, 32, 32, 256)  65792       activation_15[0][0]
________________________________________________________________________________________
conv2d_18 (Conv2D)              (None, 32, 32, 256)  65792       activation_5[0][0]
________________________________________________________________________________________
batch_normalization_13 (BatchNo (None, 32, 32, 256)  1024        conv2d_17[0][0]
```

Fig. 3 (continued)



```
________________________________________________________________________________________________
batch normalization 14 (BatchNo (None, 32, 32, 256)  1024        conv2d 18[0][0]
________________________________________________________________________________________________
activation_16 (Activation)      (None, 32, 32, 256)  0           add_1[0][0]
________________________________________________________________________________________________
conv2d_19 (Conv2D)              (None, 32, 32, 1)    257         activation_16[0][0]
________________________________________________________________________________________________
batch_normalization_15 (BatchNo (None, 32, 32, 1)    4           conv2d_19[0][0]
________________________________________________________________________________________________
activation_17 (Activation)      (None, 32, 32, 1)    0           batch_normalization_15[0][0]
________________________________________________________________________________________________
multiply_1 (Multiply)           (None, 32, 32, 256)  0           activation_5[0][0]
                                                                 activation_17[0][0]
________________________________________________________________________________________________
concatenate_1 (Concatenate)     (None, 32, 32, 512)  0           activation_15[0][0]
                                                                 multiply_1[0][0]
________________________________________________________________________________________________
conv2d_20 (Conv2D)              (None, 32, 32, 256)  1179904     concatenate_1[0][0]
________________________________________________________________________________________________
activation_18 (Activation)      (None, 32, 32, 256)  0           conv2d_20[0][0]
________________________________________________________________________________________________
conv2d_21 (Conv2D)              (None, 32, 32, 256)  590080      activation_18[0][0]
________________________________________________________________________________________________
activation_19 (Activation)      (None, 32, 32, 256)  0           conv2d_21[0][0]
________________________________________________________________________________________________
up_sampling2d_2 (UpSampling2D)  (None, 64, 64, 256)  0           activation_19[0][0]
________________________________________________________________________________________________
conv2d_22 (Conv2D)              (None, 64, 64, 128)  295040      up_sampling2d_2[0][0]
________________________________________________________________________________________________
batch_normalization_16 (BatchNo (None, 64, 64, 128)  512         conv2d_22[0][0]
________________________________________________________________________________________________
activation_20 (Activation)      (None, 64, 64, 128)  0           batch_normalization_16[0][0]
________________________________________________________________________________________________
conv2d_23 (Conv2D)              (None, 64, 64, 128)  16512       activation_20[0][0]
________________________________________________________________________________________________
conv2d_24 (Conv2D)              (None, 64, 64, 128)  16512       activation_3[0][0]
________________________________________________________________________________________________
batch_normalization_17 (BatchNo (None, 64, 64, 128)  512         conv2d_23[0][0]
________________________________________________________________________________________________
batch_normalization_18 (BatchNo (None, 64, 64, 128)  512         conv2d_24[0][0]
________________________________________________________________________________________________
add_2 (Add)                     (None, 64, 64, 128)  0           batch_normalization_17[0][0]
                                                                 batch_normalization_18[0][0]
________________________________________________________________________________________________
activation_21 (Activation)      (None, 64, 64, 128)  0           add_2[0][0]
________________________________________________________________________________________________
conv2d_25 (Conv2D)              (None, 64, 64, 1)    129         activation_21[0][0]
________________________________________________________________________________________________
batch_normalization_19 (BatchNo (None, 64, 64, 1)    4           conv2d_25[0][0]
________________________________________________________________________________________________
activation_22 (Activation)      (None, 64, 64, 1)    0           batch_normalization_19[0][0]
________________________________________________________________________________________________
multiply_2 (Multiply)           (None, 64, 64, 128)  0           activation_3[0][0]
                                                                 activation_22[0][0]
________________________________________________________________________________________________
concatenate_2 (Concatenate)     (None, 64, 64, 256)  0           activation_20[0][0]
                                                                 multiply_2[0][0]
________________________________________________________________________________________________
conv2d_26 (Conv2D)              (None, 64, 64, 128)  295040      concatenate_2[0][0]
________________________________________________________________________________________________
activation_23 (Activation)      (None, 64, 64, 128)  0           conv2d_26[0][0]
________________________________________________________________________________________________
conv2d_27 (Conv2D)              (None, 64, 64, 128)  147584      activation_23[0][0]
________________________________________________________________________________________________
activation_24 (Activation)      (None, 64, 64, 128)  0           conv2d_27[0][0]
```

Fig. 3 (continued)



```
__________________________________________________________________________________________________
up_sampling2d_3 (UpSampling2D)  (None, 128, 128, 128 0           activation_24[0][0]
__________________________________________________________________________________________________
conv2d_28 (Conv2D)              (None, 128, 128, 64) 73792       up_sampling2d_3[0][0]
__________________________________________________________________________________________________
batch_normalization_20 (BatchNo (None, 128, 128, 64) 256         conv2d_28[0][0]
__________________________________________________________________________________________________
activation_25 (Activation)      (None, 128, 128, 64) 0           batch_normalization_20[0][0]
__________________________________________________________________________________________________
conv2d_29 (Conv2D)              (None, 128, 128, 64) 4160        activation_25[0][0]
__________________________________________________________________________________________________
conv2d_30 (Conv2D)              (None, 128, 128, 64) 4160        activation_1[0][0]
__________________________________________________________________________________________________
batch_normalization_21 (BatchNo (None, 128, 128, 64) 256         conv2d_29[0][0]
__________________________________________________________________________________________________
batch_normalization_22 (BatchNo (None, 128, 128, 64) 256         conv2d_30[0][0]
__________________________________________________________________________________________________
add_3 (Add)                     (None, 128, 128, 64) 0           batch_normalization_21[0][0]
                                                                 batch_normalization_22[0][0]
__________________________________________________________________________________________________
activation_26 (Activation)      (None, 128, 128, 64) 0           add_3[0][0]
__________________________________________________________________________________________________
conv2d_31 (Conv2D)              (None, 128, 128, 1)  65          activation_26[0][0]
__________________________________________________________________________________________________
batch_normalization_23 (BatchNo (None, 128, 128, 1)  4           conv2d_31[0][0]
__________________________________________________________________________________________________
activation_27 (Activation)      (None, 128, 128, 1)  0           batch_normalization_23[0][0]
__________________________________________________________________________________________________
multiply_3 (Multiply)           (None, 128, 128, 64) 0           activation_1[0][0]
                                                                 activation_27[0][0]
__________________________________________________________________________________________________
concatenate_3 (Concatenate)     (None, 128, 128, 128 0           activation_25[0][0]
                                                                 multiply_3[0][0]
__________________________________________________________________________________________________
conv2d_32 (Conv2D)              (None, 128, 128, 64) 73792       concatenate_3[0][0]
__________________________________________________________________________________________________
activation_28 (Activation)      (None, 128, 128, 64) 0           conv2d_32[0][0]
__________________________________________________________________________________________________
conv2d_33 (Conv2D)              (None, 128, 128, 64) 36928       activation_28[0][0]
__________________________________________________________________________________________________
activation_29 (Activation)      (None, 128, 128, 64) 0           conv2d_33[0][0]
__________________________________________________________________________________________________
conv2d_34 (Conv2D)              (None, 128, 128, 1)  65          activation_29[0][0]
```

Fig. 3 (continued)

loss. However, the proposed methods produced excellent results with better DSC score, similar to manual annotation. The various steps of the proposed technique are summarized in Fig. 5.

3 Results and Discussions

The images obtained after preprocessing of the original image are given in Fig. 6. The preprocessing involves computation of a normalized image by using Hounsfield transform and thereby selecting the pixels in the range having maximum information.

The dice coefficient (DC) has been used for evaluating and comparing the segmentation results of the models, which is defined as given in Eq. (2).

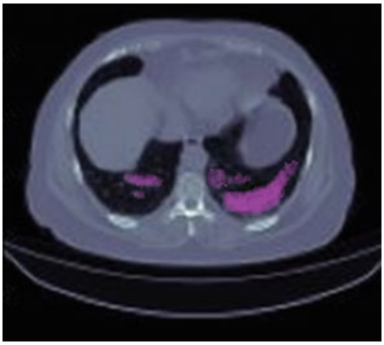
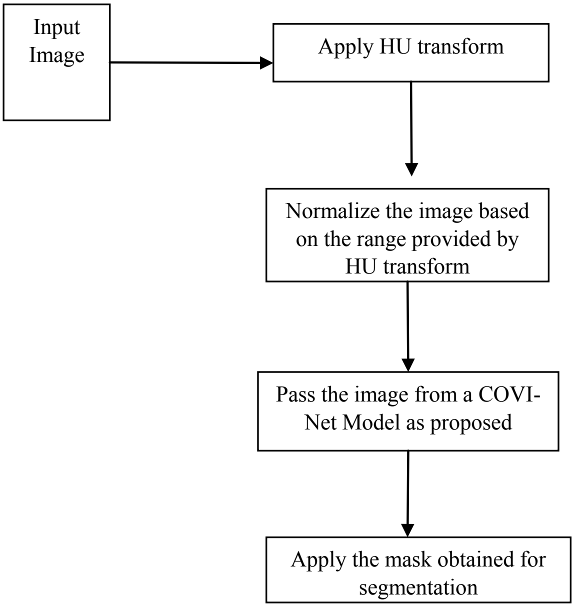


Fig. 4 Segmented image obtained by Covi-Net model



Input Image
Apply HU transform
Normalize the image based on the range provided by HU transform
Pass the image from a COVI-Net Model as proposed
Apply the mask obtained for segmentation

Fig. 5 Steps involved in the segmentation of lung CT scan images using Covi-Net model

$$\text{DSC} = \frac{2|X \cap Y|}{|X| + |Y|} \tag{2}$$

where X is the model's semantic segmentation output and Y is an expert-specified mask of infected regions. The DSC scale runs from 0 to 1, with 1 indicating the highest degree of similarity between the model output (prediction) and the truth ground. Table 1 gives the DSC and cross-entropy loss acquired for each model for the testing set. According to these findings, Covi-Net outperformed the other models, indicating that attention blocks and recurring routes may be important in detecting lesions.

The loss decreases with each epoch, is shown in Fig. 7. The appropriate training of the model is depicted by a constant decrease in loss value with each epoch and a flat

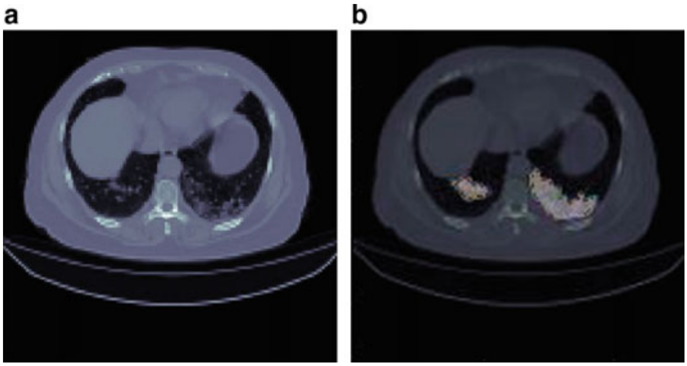


Fig. 6 **a** Original image and **b** normalized image

Table 1 Comparison of Covi-NET model with existing similar achitectures

Model	DSC
U-Net [36]	0.77
R2 U-Net [37]	0.76
Attention U-Net [38]	0.77
Attention R2 U-Net	0.79
Covi-Net	0.88

stabilizing curve at the conclusion. Figure 7 shows that the loss function's minimal value has been reached.

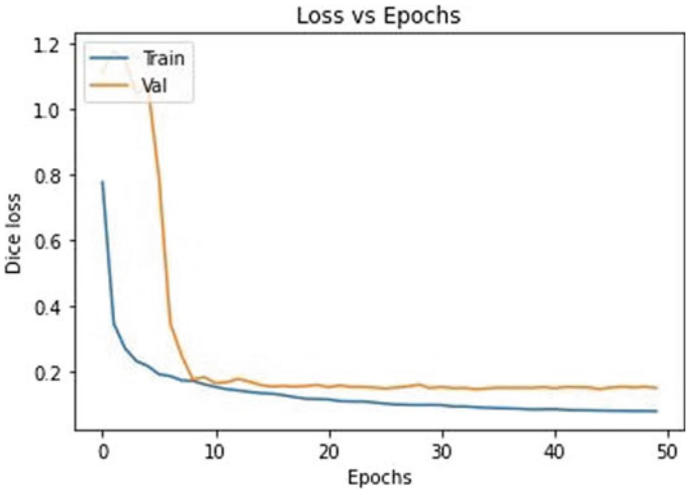


Fig. 7 Dice loss with each epoch during training of segmentation model

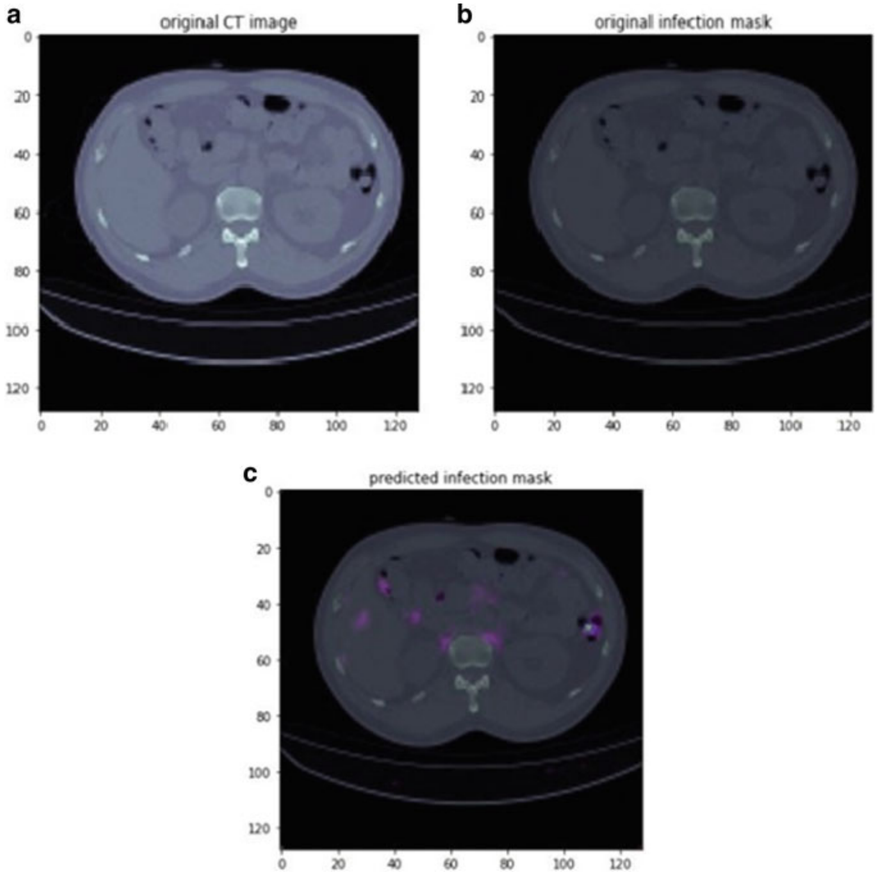


Fig. 8 **a** Orignal CT image, **b** original infection mask, and **c** predicted infection mask by Covi-Net

The prediction of lesion volumes to CT volumes has been tested to further evaluate the model's accuracy and compare the segmentation findings to the ground truth. The results in Fig. 8 reveal that the lesion volumes recognized by the model are strongly correlated with those discovered by the expert, indicating that the proposed model has a high power of prediction. The lesions have been highlighted with a purple marker and indicates the damage done due to COVID-19.

4 Conclusion and Future Scope

As observed from Table 1, the proposed Covi-Net model outperforms other state-of-the-art techniques. It has improved the DSC value for segmented images by 12.5%, 13.6%, 12.5%, and 10.2% over U-Net, R2-U-Net, Attention U-Net, and Attention R2-U-Net, respectively.

The proposed CNN framework has been obtained by fine-tuning U-Net's existing architecture. Similarly, other CNN networks like Alex-Net, VGG Models, Google-Net, etc., can be fine-tuned to achieve better segmentation models.

References

1. Bedford J, Enria D, Giesecke J, Heymann DL, Ihekweazu C, Kobinger G, Wieler LH (2020) Living with the COVID-19 pandemic: act now with the tools we have. The Lancet 396(10259):1314–1316
2. Silva ALOD, Moreira JC, Martins SR (2020) COVID-19 and smoking: a high-risk association. Cad Saude Publica 36:e00072020
3. Bai HX, Hsieh B, Xiong Z, Halsey K, Choi JW, Tran TML, Liao WH (2020) Performance of radiologists in differentiating COVID-19 from non-COVID-19 viral pneumonia at chest CT. Radiology 296(2):E46–E54
4. Ai T, Yang Z, Hou H, Zhan C, Chen C, Lv W, Xia L (2020) Correlation of chest CT and RT-PCR testing for coronavirus disease 2019 (COVID-19) in China: a report of 1014 cases. Radiology 296(2):E32–E40
5. Fang Y, Zhang H, Xie J, et al (2020) Sensitivity of chest CT for COVID-19: comparison to RT-PCR. Radiology, 200432
6. Wang D, Hu B, Hu C, Zhu F, Liu X, Zhang J, Peng Z (2020) Clinical characteristics of 138 hospitalized patients with 2019 novel coronavirus–infected pneumonia in Wuhan, China. Jama 323(11):1061–1069
7. Yan Q, Zhang L, Liu Y, Zhu Y, Sun J, Shi Q, Zhang Y (2020) Deep HDR imaging via a non-local network. IEEE Trans Image Process 29:4308–4322
8. Yan Q, Gong D, Zhang Y (2018) Two-stream convolutional networks for blind image quality assessment. IEEE Trans Image Process 28(5):2200–2211
9. Yan Q, Gong D, Zhang P, Shi Q, Sun J, Reid I, Zhang Y (2019) Multi-scale dense networks for deep high dynamic range imaging. In: 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), pp 41–50, January. IEEE
10. Yan Q, Gong D, Shi Q, Hengel AVD, Shen C, Reid I, Zhang Y (2019) Attention-guided network for ghost-free high dynamic range imaging. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 1751–1760
11. Gong D, Yang J, Liu L, Zhang Y, Reid I, Shen C, et al (2017) From motion blur to motion flow: A deep learning solution for removing heterogeneous motion blur. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2319–2328
12. He T, Shen C, Tian Z, Gong D, Sun C, Yan Y (2019) Knowledge adaptation for efficient semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 578–587
13. Gong D, Liu L, Le V, Saha B, Mansour MR, Venkatesh S, Hengel AVD (2019) Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 1705–1714
14. Wang L, Lin ZQ, Wong A (2020) Covid-net: a tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images. Sci Rep 10(1):1–12
15. Wang S, Kang B, Ma J, Zeng X, Xiao M, Guo J, et al (2021) A deep learning algorithm using CT images to screen for Corona Virus Disease (COVID-19). European Radiol, 1–9
16. Joaquin AS, To detect pneumonia caused by NCOV- 19 from x-ray images. https://towardsdatascience.com/using-deep-learning-to-detect-ncov-19-from-x-ray-images-1a89701d1acd. Accessed November 2, 2021

17. Chowdhury MEH, Rahman T, Khandakar A, Mazhar R, Kadir MA, Mahbub ZB, Islam KR et al (2020) Can AI help in screening viral and COVID-19 pneumonia? IEEE Access 8:132665–132676
18. Gozes O, Frid-Adar M, Greenspan H, Browning PD, Zhang H, Ji W, Siegel E (2020) Rapid Al development cycle for the coronavirus (covid-19) pandemic: Initial results for automated detection & patient monitoring using deep learning CT image analysis. arXiv preprint arXiv:2003.05037
19. Tang Z, Zhao W, Xie X, Zhong Z, Shi F, Liu J, Shen D (2020) Severity assessment of coronavirus disease 2019 (COVID-19) using quantitative features from chest CT images. arXiv preprint arXiv:2003.11988
20. Shi F, Xia L, Shan F, Song B, Wu D, Wei Y, Shen D (2021) Large-scale screening to distinguish between COVID-19 and community-acquired pneumonia using infection size-aware classification. Phys Med Biol 66(6):065031
21. Shan F, Gao Y, Wang J, Shi W, Shi N, Han M, et al (2020) Lung infection quantification of COVID-19 in CT images with deep learning. arXiv preprint arXiv:2003.04655
22. Apostolopoulos ID, Mpesiana TA (2020) Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks. Phys Eng Sci Med 43(2):635–640
23. El Asnaoui K, Chawki Y, Idri A (2021) Automated methods for detection and classification pneumonia based on x-ray images using deep learning. In: Artificial intelligence and blockchain for future cybersecurity applications. Springer, Cham, pp 257–284
24. Xu X, Jiang X, Ma C, Du P, Li X, Lv S, Li L (2020) A deep learning system to screen novel coronavirus disease 2019 pneumonia. Engineering 6(10):1122–1129
25. Hemdan EED, Shouman MA, Karar ME (2020) Covid x-net: a framework of deep learning classifiers to diagnose covid-19 in x-ray images. arXiv preprint arXiv:2003.11055
26. Department of Biomedical Engineering, Zonguldak Bulent Ecevit University, 67100, Zonguldak, Turkey
27. Barstugan M, Ozkaya U, Ozturk S (2020) Coronavirus (covid-19) classification using CT images by machine learning methods. arXiv preprint arXiv:2003.09424
28. Sethy PK, Behera SK (2020) Detection of coronavirus disease (covid-19) based on deep features
29. Ismael AM, Şengür A (2021) Deep learning approaches for COVID-19 detection based on chest X-ray images. Expert Syst Appl 164:114054
30. Khan AI, Shah JL, Bhat MM (2020) CoroNet: A deep neural network for detection and diagnosis of COVID-19 from chest x-ray images. Comput Methods Programs Biomed 196:105581
31. Brunese L, Mercaldo F, Reginelli A, Santone A (2020) Explainable deep learning for pulmonary disease and coronavirus COVID-19 detection from X-rays. Comput Methods Programs Biomed 196:105608
32. Maghdid HS, Asaad AT, Ghafoor KZ, Sadiq AS, Mirjalili S, Khan MK (2021) Diagnosing COVID-19 pneumonia from X-ray and CT images using deep learning and transfer learning algorithms. In: Multimodal Image Exploitation and Learning (vol 11734). International Society for Optics and Photonics, p. 117340E
33. Gupta A, Gupta S, Katarya R (2021) InstaCovNet-19: A deep learning classification model for the detection of COVID-19 patients using Chest X-ray. Appl Soft Comput 99:106859
34. Bharati S, Podder P, Mondal M, Prasath VB (2021) Medical imaging with deep learning for COVID-19 diagnosis: a comprehensive review. arXiv preprint arXiv:2107.09602
35. Yang X, He X, Zhao J, Zhang Y, Zhang S, Xie P (2020) COVID-CT-dataset: a CT scan dataset about COVID-19. arXiv preprint arXiv:2003.13865
36. Chen X, Yao L, Zhang Y (2020) Residual attention u-net for automated multi-class segmentation of covid-19 chest CT images. arXiv preprint arXiv:2004.05645
37. Zhou T, Canu S, Ruan S (2021) Automatic COVID-19 CT segmentation using U-Net integrated spatial and channel attention mechanism. Int J Imaging Syst Technol 31(1):16–27
38. Oktay O, Schlemper J, Folgoc LL, Lee M, Heinrich M, Misawa K, Rueckert D (2018) Attention u-net: learning where to look for the pancreas. arXiv preprint arXiv:1804.03999

A Novel Approach to Recognize Optical Characters of Number Plate Using Object Detection

Arnav Bhardwaj, Kaivalya Sourav Srivastava, Antara Kar, and Sumita Gupta

Abstract Today, the world has enhanced in terms of security but still there are occasions where people do not change even with the changing times. In the recent scenario, cars are replete and with amelioration in the number of cars, the susceptibility to threats also has increased dramatically. A complex, precise and high performing software is needed to detect the number plate to keep a check on the security and the traffic. For such surveillance work, it is paramount that recent computer domains like object detection, object recognition, image segmentation, optical character recognition and other domains are used to develop a solution that gives high accuracy based on the given available conditions. Detecting and recognizing the number plates and enhancing the accuracy in which the license numbers of cars are detected are the need of the hour. In this paper, two approaches to detect number plates are compared, which are, performing optical character recognition (OCR) on original images, and performing object detection on the images to locate cars and then performing OCR on the cropped images of the cars. The considered OCR models are EasyOCR, Keras-OCR, WPOD-Net as well as convolution neural network (CNN) and You Only Look Once (YOLOv3) model for object detection. The results of the OCR models were compared with the actual texts using Levenshtein distance. On the basis of the performance, EasyOCR performs best, providing a similarity ratio of 71.8% on original images, which further improves to 81.2%, when combined with YOLOv3 object detection.

Keywords Optical character recognition · Object detection · Number plate text recognition · You only look once

A. Bhardwaj (✉) · K. S. Srivastava · A. Kar · S. Gupta
Department of Computer Science & Engineering, ASET, Amity University, Noida, Uttar Pradesh, India
e-mail: arnavbhardwaj622@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_61

1 Introduction

Human beings have the capability to recognize objects in the surroundings and determine their importance in fractions of seconds. The brain has the capability to detect those parts of the scenario that are relevant and remember them for the future or for performing tasks in hand. Computer programmes are being created to replicate the complex calculations that human brains can do using artificial intelligence. One of the applications of this artificial intelligence is performing text recognition such as seen in Google Cloud Vision API [1].

Machine learning is nothing but a part of artificial intelligence that emphasizes on making the machine learn patterns in the data and then perform analysis on unseen data with that learning. The learning process is basically creating computing algorithms that take in the data and determine the patterns in that data. This learning helps in analysing large amounts of data, recognizing patterns in the data and predicting the outputs in certain scenarios. With the advancement in machine learning and introduction of fields like deep learning, this approach can be extended to a degree, where patterns in images can be determined and objects of interest are found and object detection is performed for cars, humans, animals, etc., and optical character recognition performed for text as well.

In this paper, number plate text recognition is performed using four models, which are EasyOCR, Keras-OCR, convolution neural network (CNN) and warped planar object detection network (WPOD-Net). These models have been taken for understanding their performance at detecting text from images that contain number plates and their performance at recognizing characters individually. OCR has been performed using these models. The generated texts and the actual expected texts are compared henceforth. The dataset was collected by taking pictures of cars from real life, thereby checking the performance of the models thoroughly, as the data is completely new to the models.

To go a step further in the research, object detection model, You Only Look Once (YOLOv3 model), has been used to segregate the parts where the car is present from the rest of the image, and then, once again performed OCR on those cropped images. The findings show how performing object detection and then OCR for number plate text recognition improves the results when compared to simply performing OCR on the images.

2 Related Works

Panhwar et al. [2] have put forth a model that takes the images from environment through a smart device and detects the text in a signboard, if it is present in the image. Images are scanned for signboards, text (English and Urdu) extracted using ANN for OCR and it gives 85% accuracy. Shi et al. [3] have proposed a CRNN in their paper which does better than other scene text detection. The model consists of convolution layers at the bottom, followed by recurrent layers in the middle, which are then connected to the transcription layer on the top. They have been able to achieve 97.6% accuracy on the IIT5k dataset, outperforming most of the existing models.

As per Goodfellow et al. [4], they have proposed a DCNN that uses the DistBelief architecture [5] for training the model on the images directly. They have stated that with increase in the depth of their architecture, the performance of the model increased, ultimately they reached 11 layers to get best performance and thus an accuracy of 96% on SVHN dataset and reportedly an accuracy of 99.8% on the hardest reCAPTCHA dataset. Zhou et al. [6] provide a very fast and light weighted model EAST detector that uses a single layer of neural network for determining the text from images surpassing many state-of-the-art implementations of recognizing the text in a scene. A model which needs small training time, utilizing the loss functions, achieved an f-score of 78.2% with a frame rate of 13.2 fps at 720p resolution.

Yadav et al. [7] have used RNN and LSTM for character recognition. For the image processing, they perform binary transformation, noise reduction and skew correction and finally, generate the feature vector of the image to be sent for classification. The model consisting of 128 hidden layers, two neural layers and ten epochs was accurate up to 98.46%. Wojna et al. [8] have proposed architecture that works on attention-based extraction (ABE) mechanism. The model is based on CNN, RNN and ABE and achieves an accuracy of 84.2%, surpassing previous models applied on French street name signs (FSNS), being better in accuracy, size and speed.

Mnih et al. [9] give a computationally light model which is compared to the existing models of neural network for scene text recognition. They realized that increased pixels increase computation time. It is similar to a neural network, having a distinction that they look for regions in the image having text; the model performs with a minimum error rate of 1.29%. Bartz et al. [10], in their paper, have proposed a spatial transformer network (STN), that is a semi-supervised model. Offering only a single, along with semi-supervision, can itself learn to detect the portions of an image containing text and recognize the text as well. The proposed model gives an accuracy of 95.2%, which is at par with other existing models.

Jaderberg et al. [11] have introduced a new model for spatially manipulating data in the network. The model without modifications and optimizations can be inserted into other models for spatially transforming the data. Their work shows promising results with feed forward architecture and RNN. They claim the model to be a surpassing other models in performance at various benchmarks. Srivastava and Gupta [12] have done lane detection by improving the OpenCV model and adding canny edge detection to it. Goel et al. [13] in their paper have designed a new model that performs holistic word recognition. The model, weighted dynamic time warping (wDTW), performs text recognition without performing text segmentation or binary conversion of the image. They use gradient-based features for representing the images from the scene, followed by the use of their novel approach, with which they perform text recognition on image with synthetic image features. Their model outperforms many earlier proposed models at various benchmarks datasets.

Michael et al. [14] in their paper present a general trainable framework to detect objects in cluttered scenes. The system gets knowledge from examples, not from prepared models. The base system performs with an accuracy of 43.5%; however, the extended version performs at 53.8% accuracy; false positive rates are 1:236,500 and 1:90,000, respectively.

Xuangeng et al. [15] in their paper have proposed a method to detect highly overlapped instances in crowded places. In crowded places, instead of individually distinguishing the objects, nearby overlapping objects are going to infer the same set of instances. The five proposed models came up with averaged precision accuracy of over 86%, miss rate ratio of around 0.4 and Jaccard index of around 0.8.

3 Optical Character Recognition and Object Detection Techniques

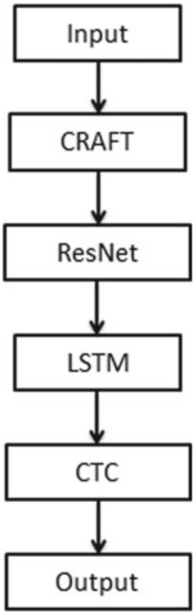
Optical character recognition (OCR) refers to the conversion of textual data captured in images into written text that can be edited through machines. OCR involves various disciplines including image processing, recognizing patterns and computer vision. The emphasis on OCR has increased in recent times as the data can be easily stored in electronic form, when compared to the actual physical space required to store textual data.

3.1 EasyOCR

As the name suggests, EasyOCR is a Python tool that can very easily perform OCR on images for Python developers. Also, EasyOCR is one of the most straightforward methods for OCR. This package is created by Jaided AI who have a specialty in creating OCR tools. This library is implemented using PyTorch library and currently supports more than 58 languages including English, Hindi, Marathi, Bangla and many more.

The framework consists of the following layers, character region awareness for text detection (CRAFT) [16], which finds the text in the image and creates the bounding boxes for them. Residual neural network (ResNet) [17] has been used for training on the dataset comprising of characters. LSTM [18] and CTC [19] layers have been added to improve the model as they can process the entire sequence of data rather than a single input. The framework has been depicted in Fig. 1.

Fig. 1 EasyOCR framework

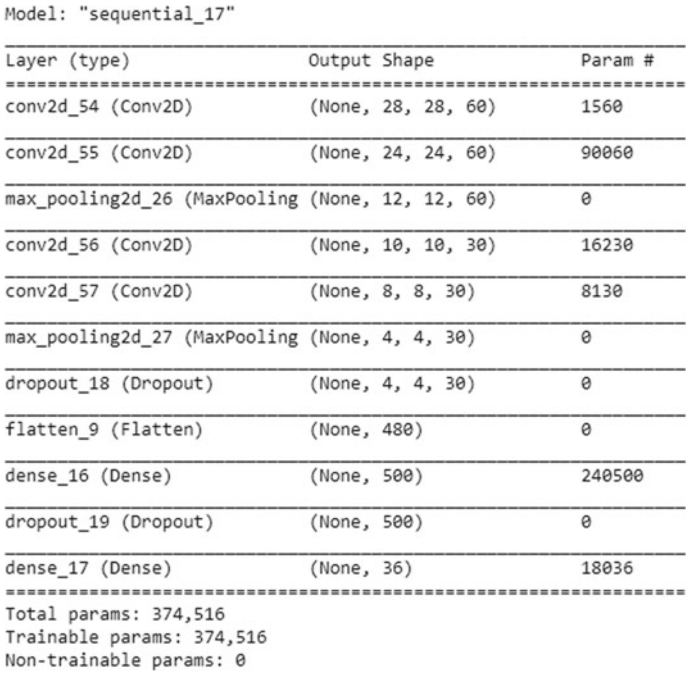


3.2 Keras-OCR

Keras-OCR is based upon the CRNN text recognition and CRAFT text detection model. It provides an OCR pipeline for converting text in images into digital text. Keras-OCR has proven itself to be working very well on scene text detection, the most required aspect for number plate text detection in our research. The framework consists of the CRAFT [16] text detection model followed by CRNN model for text recognition.

3.3 Convolution Neural Network (CNN)

This model consists of the Haar cascade model [20] for detecting number plates, trained on Indian number plates, followed by a convolution neural network that has been trained on English letters and numbers. The model takes the images of English characters and numbers with a resolution of 28 × 28 pixels, goes through two 2D convolution layers followed by a MaxPooling layer, followed by two more layers of 2D convolution and a MaxPooling layer. Two layers of dropout have also been used for preventing overfitting of the data. The CRNN model summary is shown in Fig. 2.



```
Model: "sequential_17"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_54 (Conv2D)           (None, 28, 28, 60)        1560
_________________________________________________________________
conv2d_55 (Conv2D)           (None, 24, 24, 60)        90060
_________________________________________________________________
max_pooling2d_26 (MaxPooling (None, 12, 12, 60)        0
_________________________________________________________________
conv2d_56 (Conv2D)           (None, 10, 10, 30)        16230
_________________________________________________________________
conv2d_57 (Conv2D)           (None, 8, 8, 30)          8130
_________________________________________________________________
max_pooling2d_27 (MaxPooling (None, 4, 4, 30)          0
_________________________________________________________________
dropout_18 (Dropout)         (None, 4, 4, 30)          0
_________________________________________________________________
flatten_9 (Flatten)          (None, 480)               0
_________________________________________________________________
dense_16 (Dense)             (None, 500)               240500
_________________________________________________________________
dropout_19 (Dropout)         (None, 500)               0
_________________________________________________________________
dense_17 (Dense)             (None, 36)                18036
=================================================================
Total params: 374,516
Trainable params: 374,516
Non-trainable params: 0
```

Fig. 2 CRNN model summary

3.4 Warped Planar Object Detection Network (WPOD-Net)

This model consists of the warped planar object detection network [21] as the number plate detection model along with the MobileNets [22] character recognition model for character recognition of number plate numbers. The network architecture for WPOD-Net is shown in Fig. 3.

3.5 You Only Look Once (YOLOv3)

The YOLO model treats the entire image by putting it under a neural network possessing a single layer. This network thereafter divides the image into areas, discretely separating them by putting bounding boxes. The fact that only a single layer is applied on the image makes this model quicker than Fast R-CNN. The YOLO model specially suffered from small objects, thus it suffered on the accuracy front. YOLOv3 is an updated version of the YOLO model which though has a larger size but is more accurate and has also kept the running time intact. YOLOV3 which is also known as You Only Look Once, is a prominent object detection model architecture which can be used to identify specific objects in videos, images and live feeds. In the

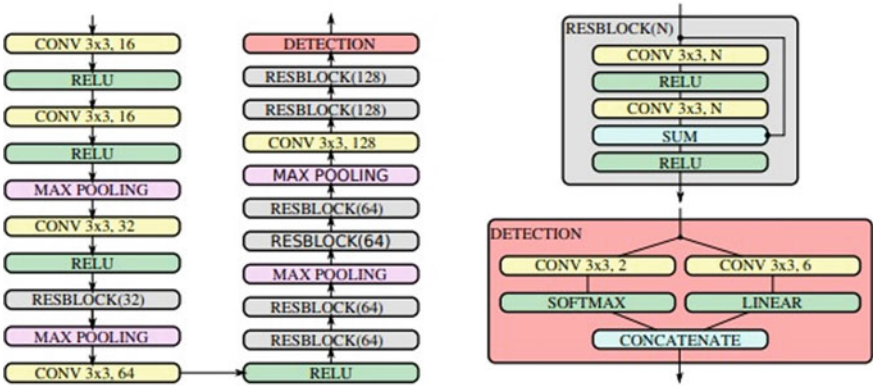


Fig. 3 WPOD-Net network architecture [21]

object detection architecture, convolutional neural network architecture is used here. Unlike other neural network architecture, the specialty of YOLO lies in the fact that it is based on a 1*1 convolutions convolution layer. The benefit of YOLO was that it is much faster than traditional object detection techniques without compromising on the accuracy.

4 Methodology

The research commences with applying all the above mentioned optical character recognition models, viz., EasyOCR, Keras-OCR, CNN, WPOD-Net and analysed their performance on the collected dataset, by using the concept of Levenshtein distance [23] which tells how many edits are required to make the two given strings equal. Also, the average ratio of similarity was determined by finding the sum of lengths of the strings, subtracting the number of edits and then dividing the whole by the sum of lengths of the strings.

4.1 Proposed Architecture

It was inferred that EasyOCR model stands out among the four models. However, the performance was not as good as expected, therefore, objection detection was brought in to enhance the model, so as to achieve a higher performance on our OCR models. After working through the objection detection models, YOLOv3 model stood out as the best performer and its results combined with EasyOCR gave even more promising results. Figure 4 presents the architecture proposed.

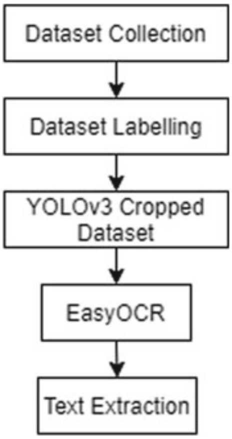


Fig. 4 Architecture proposed for number plate text recognition

4.2 Implementation

The basic understanding of the flow of events in our research is as follows: The dataset is collected and labelled as per the number plates in the images. After labelling is done, the four said OCR methods were performed on those images (original images) and the results were stored. After this, YOLOv3 object detection was performed on these images using YOLOv3 model to get only the part of images (cropped images) where a car was present. Further, once again the four OCR models were applied on these cropped images and the results were stored. The results are followed by a comparative analysis. Figure 5 shows the process that was used for determining the best architecture for number plate text recognition.

After the cropped images are generated, it was once again needed to label the images as per their number plates, as it was seen that the number of generated cropped images was more than original images, which happened due to multiple cars in some images. There were 131 images generated from the YOLOv3 model, which were then labelled and fed to the OCR models. For example, Fig. 5 shows the original image, where the detected text with EasyOCR was "Get Instant", and after cropping the image with the help of YOLOv3, the output generated was as desired, as shown in Fig. 6.

5 Results and Analysis

To measure the performance yielded by these models, authors have created their own dataset by taking images of moving cars and cars parked on the roadside. The dataset contains 127 images of cars, which have been taken through a phone camera with a resolution of 96dpi and dimensions of 2250 × 4000 pixels. The images offer a wide range of cars and various positions in which the number plates are present. Once the

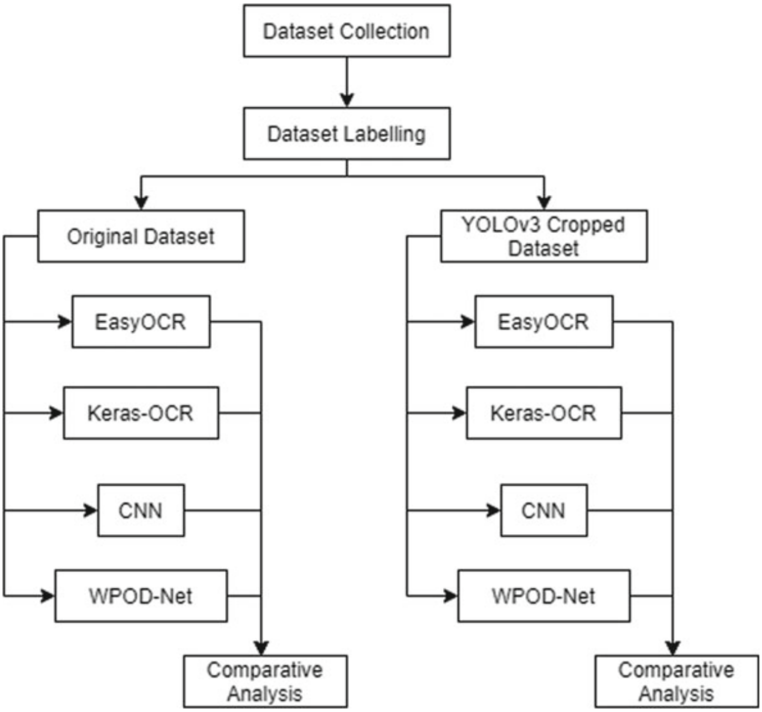


Fig. 5 Flow of events to reach the proposed architecture

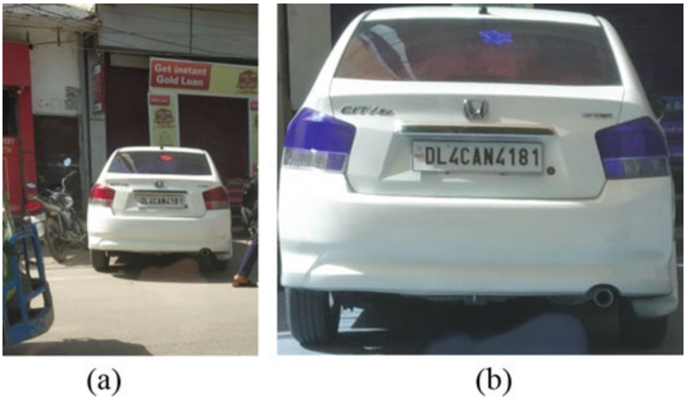


Fig. 6 **a** OCR on original image (output with EasyOCR: 'Get Instant'), **b** OCR on YOLOv3 cropped image (output with EasyOCR: actual license number)

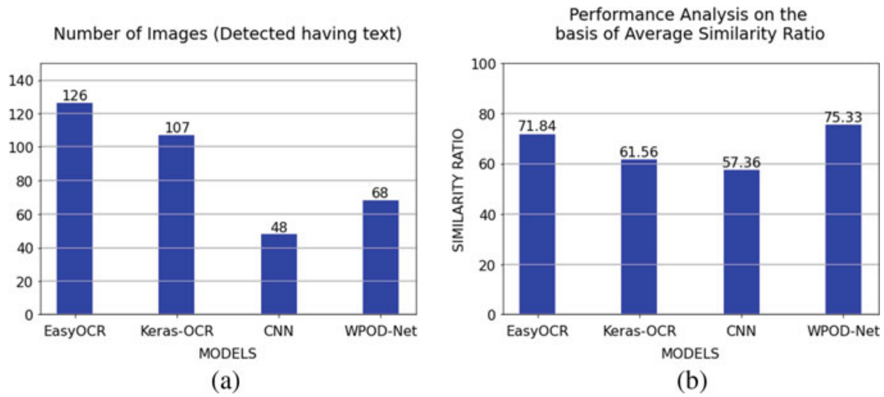


Fig. 7 **a** Number of images detected having text by respective models and **b** performance analysis on the basis of average similarity ratio

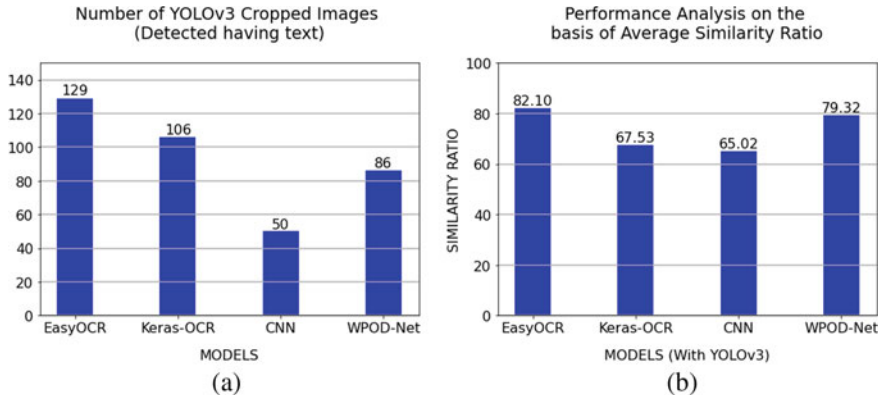
dataset was collected, labelling of the dataset was done by taking all the images and storing the number plate numbers in another file, for retrieving later in the research to compare with and analyse the outputs given by the models.

OCR models perform well on the images that were taken as the dataset. There were a total 127 original images. The results of the OCR models can be visualized from the graphical representation in given in Fig. 7. Figure 7a contains the count of original images that the models could detect text in them and Fig. 7b contains the average similarity ratio of all these models on the original images.

After performing OCR on the original images, the images were cropped using YOLOv3, generating 131 images. The results of the OCR models on the YOLOv3 cropped images are shown in Fig. 8. In the following Fig. 8a, the number of images that were detected to contain text in them and Fig. 8b provides the average similarity ratio of all these models on the YOLOv3 cropped images.

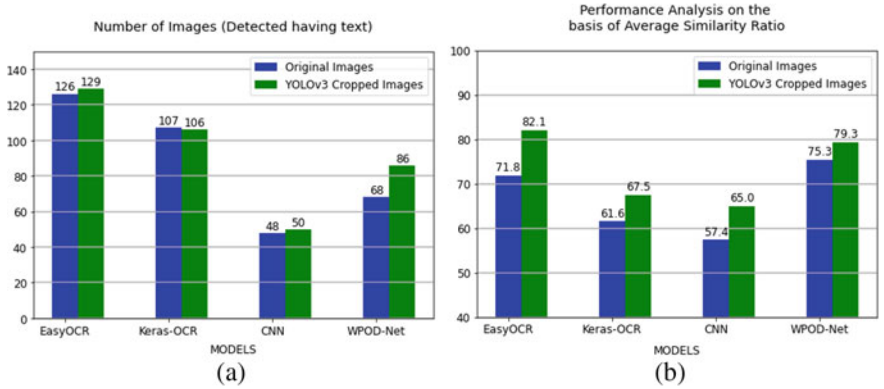
Figure 9a and b shows the results generated when the images were subjected to OCR without being cropped (original images) and when they were subjected to OCR after cropping them with YOLOv3, to get only the portion where the cars were present. Figure 9a compares the number of images that were found having text with the four OCR models with and without cropping. Figure 9b shows the comparison of average similarity ratios of the OCR models before and after cropping.

The four OCR models were implemented on our own dataset, and the performance of the models analysed on the basis of average similarity ratio came out to be 75.33% for WPOD-Net, 71.84% for EasyOCR, 61.56% for Keras-OCR and 57.36% for CNN, when implemented on original dataset and 79.32% for WPOD-Net, 82.10% for EasyOCR, 67.53% for Keras-OCR and 65.02% for CNN, when implemented on images cropped with YOLOv3.



(a) (b)

Fig. 8 **a** Number of YOLOv3 cropped images detected having text in them with respective models and **b** performance analysis on the basis of average similarity ratio



(a) (b)

Fig. 9 **a** Comparative analysis of number of images detected having text with and without cropping and **b** performance comparison on the basis of average similarity ratio

6 Conclusion

In the research, it was seen that the accuracy of the OCR models improved with the application of the object detection model. Therefore object detection was brought in to crop the images to only required parts and thus boost the performance of OCR models. The images cropped by the YOLOv3 model were fed to each of the OCR models, and an improvement is witnessed in the performance of all the OCR models due to this. The most accurate results were noticed when YOLOv3 was used in combination with EasyOCR, for object detection and character recognition, respectively.

Further, improvements to the work presented in the paper can be accomplished using YOLOv5 and its other newer improved versions or other object detection models by designing better trained models for robust performance on challenging images.

References

1. Walker J, Fujii Y, Popat AC (2018) A web-based ocr service for documents. In: Proceedings of the 13th IAPR international workshop on document analysis systems (DAS), Vienna, Austria vol. 1
2. Panhwar MA, Memon KA, Abro A, Zhongliang D, Khuhro SA, Memon S (2019) Signboard detection and text recognition using artificial neural networks. In: 2019 IEEE 9th international conference on electronics information and emergency communication (ICEIEC), pp 16–19. IEEE
3. Shi B, Bai X, Yao C (2016) An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE Trans Pattern Anal Mach Intell 39(11):2298–2304
4. Goodfellow IJ, Bulatov Y, Ibarz J, Arnoud S, Shet V (2013) Multi-digit number recognition from street view imagery using deep convolutional neural networks. arXiv preprint arXiv:1312.6082
5. Dean J, Corrado G, Monga R, Chen K, Devin M, Mao M, Ranzato MA, Senior A, Tucker P, Yang K, Le Q (2012) Large scale distributed deep networks. Adv Neural Inf Process Syst 25:1223–1231
6. Zhou X, Yao C, Wen H, Wang Y, Zhou S, He W, Liang J (2017) East: an efficient and accurate scene text detector. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5551–5560
7. Yadav S, Pandey A, Aggarwal P, Garg R, Aggarwal V Handwriting recognition using LSTM networks. Int J New Technol Res 4(3):263101
8. Wojna Z, Gorban AN, Lee DS, Murphy K, Yu Q, Li Y, Ibarz J (2017) Attention-based extraction of structured information from street view imagery. In 2017 14th IAPR international conference on document analysis and recognition (ICDAR) vol 1, pp 844–850. IEEE.
9. Mnih V, Heess N, Graves A (2014) Recurrent models of visual attention. In: Advances in neural information processing systems, pp 2204–2212
10. Bartz C, Yang H, Meinel C (2017) STN-OCR: a single neural network for text detection and text recognition. arXiv preprint arXiv:1707.08831
11. Jaderberg M, Simonyan K, Zisserman A (2015) Spatial transformer networks. Adv Neural Inf Process Syst 28:2017–2025
12. Srivastava S, Gupta S (2021) Path detection for self-driving carts by using canny edge detection algorithm. In: 2021 9th international conference on reliability, infocom technologies and optimization (trends and future directions) (ICRITO), pp 1–5. IEEE
13. Goel V, Mishra A, Alahari K, Jawahar CV (2013) Whole is greater than sum of parts: Recognizing scene text words. In: 2013 12th international conference on document analysis and recognition, pp 398–402. IEEE
14. Oren M, Papageorgiou CP, Poggio T A framework for object detection
15. Chu X, Zheng A, Zhang X, Sun J Detection in crowded scenes: one proposal, multiple predictions
16. Baek Y, Lee B, Han D, Yun S, Lee H (2019) Character region awareness for text detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 9365–9374
17. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778

18. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
19. Graves A, Fernández S, Gomez F, Schmidhuber J (2006) Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In: Proceedings of the 23rd international conference on machine learning, pp 369–376
20. https://github.com/SarthakV7/AI-based-indian-license-plate-detection/blob/master/indian_license_plate.xml
21. Silva SM, Jung CR (2018) License plate detection and recognition in unconstrained scenarios. In: Proceedings of the European conference on computer vision (ECCV), pp 580–596
22. Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H (2017). Mobilenets: efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861
23. Yujian L, Bo L (2007) A normalized Levenshtein distance metric. IEEE Trans Pattern Anal Mach Intell 29(6):1091–1095

Predicting Personalities Through Online Posts Using NLP

Bhavey Malik, Varun Swaminathan, Aditya Sharma, Rachna Jain, Preeti Nagrath, and Ashish Kumar Singh

Abstract Predicting personalities could be viable in many fields, and many technological efforts are being made for the same. Here, the problem statement is to be able to forecast people's personality types based on the text they have written and also evaluate the viability of the MBTI's test and its capacity to predict language styles and behaviour using MBTIs, NLP, and machine learning algorithms, and a new method of machine learning was designed to forecast the personality type according to the MBTI. To begin with, natural language processing and personality prediction are introduced, and the previous developmental works in both the fields have been studied along with the different past approaches towards the problem. Then, the methodology of research has been discussed along with the dataset, which contains about 8600 observations, where each observation gives a person's Myers-Briggs personality type as a 4-letter code and a reading containing the last 50 posts on their social media where each entry is separated by "|||" sign. Then, results produced by different models such as Naïve Bayes and SVM, and when the new method given in this study is compared to other current methods, the results reveal that the new method is more accurate and reliable.

Keywords NLP · Personality prediction · NLTK · MBTI · SVM · Logistic

1 Introduction

Natural language processing (NLP) is a set of techniques that can be used to figure out how humans think, communicate, and act. In other words, NLP can be used to analyze human behaviour patterns. Charvet [1] described NLP as a subpart of artificial intelligence which helps us in understanding human languages to perform multiple operations that include recognition of speech, segmentation of speech. Briggs-Myers et al. [2] described the Myers Briggs indicator and the potential use of it by saying

B. Malik (✉) · V. Swaminathan · A. Sharma · R. Jain · P. Nagrath · A. K. Singh
Computer Science and Engineering, Bharati Vidyapeeth's College of Engineering, New Delhi, India
e-mail: bhavya.mlik@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_62

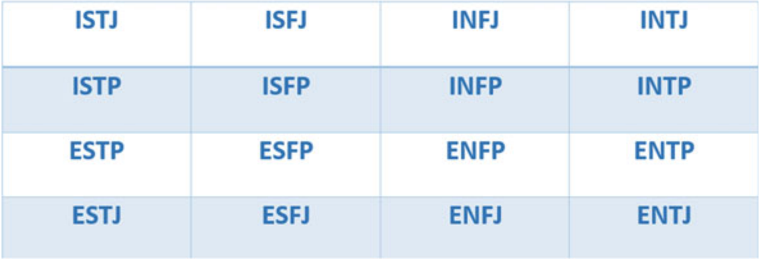
that this indicator divides people into 16 different categories each one different from each other in terms of behaviours and nature. Davis and Bigelow [3] quoted “intelligent machines” by saying there lies a huge potential of combining statistical meta-modelling like natural language processing with insights from theoretical and practical modelling and rejecting the “purist” approach of some statisticians. Lima and Leandro [4] worked on Big 5 model which is different than MBTI classification but is relevant in determining personality of a person, where the acronym used for traits was openness, conscientiousness, extraversion, agreeableness, and neuroticism (OCEAN) which extracted meta-attributes dividing them into grammatical and social behavioural category and proposes a unique system PERSOMA and predicts specific personality traits. Brian [5] explained that meta-systems are the most comprehensive strategies a person uses at all times, and they are the different ways a person can organize data. In addition, meta-systems are a popular approach to enter, filter, and filter data from the world around us, according to Davis [3]. To put it another way, it is our method of thinking, or our common techniques and patterns. Meta-systems are developed in the early phases of NLP development when Handler and Bandler collaborated [5] and discovered that people utilized different strategies to perform different things. As a result, they introduced the first list of NLP machine learning systems incorporating 60 different patterns. Wan et al. [6] showed by calculating Pearson correlation between personality dimensionality, there is a match between the words a person uses, and their personality for example neuroticism is related with words related to anger, fury, and rage and also related to extraversion. This shows high chances of calculating personality using NLP. The four standards or basic categories are extroversion-introversion (E-I), sensing-intuition (S–N), thinking-feeling (T-F), and judgement-perception (J-P).

Literature Review

Before starting this study, we looked into personality prediction using several machine learning techniques. Sharma and Kaur [7] predicted the personality through the subject’s tweets using stochastic gradient descent to train their model with promising results. Few of the concerns in the previous models were also addressed in their research, using logistic regression and a minimized error function, giving higher accuracies than Naïve Bayes. In the same year, Sultana et al. [8] analyzed the patterns of tweets in regard to four aspects like replies, hashtags, and two more. Ngatirin et al. [9] compared the different weak classifiers like trees, Bayes, and others in predicting personality into one of the big five model through Twitter statistics of students in four aspects and measuring several classification parameters also. Another dynamic model came up in the research of Berkay et al. [10] predicted personality through visual and audio cues using random forest, which gave them higher accuracies than most of the text-based models. Amirhosseini et al. [11] used natural language processing for removing the human factor from the meta-model of neuro linguistic programming and thus improve its efficiency and accuracy in personality development. In their research, Gjurković and Šnajder [12] used a Reddit dataset with MBTI types for personality prediction. Bassignana et al. [13] worked on Italian YouTube comments and labelled their types on their own and manually checked and found some error in classifying them, assuming 6–7% error for the whole dataset. They performed working on three basis-lexical, stylistically emojis handling, and embedded base but the initial assumption of error did not let the accuracy go very high. Marouf et al. [14] applied and compared several feature selection algorithms with different permutations to get the subset of best psycholinguistic features for each trait, selected the better accuracies, and made prediction using classifiers to select the best algorithm based on the big five personality traits, but since too many features are involved, it could pose a drawback due to more comparisons causing lower efficiency, and some faster elimination methods could be implemented. Abidin et al. [15] translated the language to English on which deep learning was performed after adding other features like words per comment, links per comment, punctuations which showed high correlation with the personality type, but could not explain the overall result to be less than 17% in case of SVM when the individual results were in range 50–75%. Genina et al. [16] created a model to segregate Twitter tweets into three divisions: positive, negative, and neutral. Based on these divisions, they tried to predict the personalities of a person. This was not a fool-proof method as it divided them into only 3 categories which created a huge blind spot of error to be looked at. Amirhosseini and Kazemian [17] created a new model of personality prediction using MBTI. They had used CatBoost and XGBoost algorithms in identifying the personalities. Their approach in the paper had presented great accuracy and better reliability than other machine learning algorithms. Choong and Varathan [18] founded that usage of LightGBM classifier resulted in better prediction results than other machine learning algorithms. Although some researchers provided a Kaggle dataset with a wide range of topics, it did not hide the fact that the user classes represented on the corpus are substantially imbalanced and far from the realistic distribution. Another issue with the currently accessible big corpuses was the removal of the MBTI label. Nisha and Umme [19] did not take punctuations into consideration at all and removed them in the pre-processing part and to refine them further they preferred stemming over lemmatizing and could have performed better if the dataset was more proportionate and balanced which is addressed by Kadambi [20] which changed the disproportion of different indicators through sampling and classification of four traits individually using BERT technology.

1.1 Personality Types

The word personality comes from the Latin word persona, which means "to describe an individual's behaviour or character" [2]. As mentioned above, personal preferences are divided into four sizes and 16 combinations. Figure 1 shows these 16 personality types and the type of profession they are best suited for. Introversion or extroversion characteristic is shown with "I" or "E", respectively, intuition or sensing is shown by "S" or "N". Thinking or feeling is shown by "T" or "F", and judging or perceiving character is displayed by "J" or "P", respectively.



ISTJ	ISFJ	INFJ	INTJ
ISTP	ISFP	INFP	INTP
ESTP	ESFP	ENFP	ENTP
ESTJ	ESFJ	ENFJ	ENTJ

Fig. 1 Personality types in the Myers–Briggs type indicator

1.2 Background of Automating Personality Type Prediction

Automated personality prediction using social media has a variety of uses, including résumé review, dating applications, carrier counselling, and targeting a portion of the audience by advertisement. Many human prediction studies focus on big five [4] or MBTI [13] personality models, out of which the Myers-Briggs type indicator divides personality types into 16 categories on the basis of four major types: (i) introversion(I)/extroversion(E), (ii) sensing(S)/intuition(N), (iii) thinking(T)/feeling(F), and (iv) judging(J)/perceiving(P) [11]. One letter from each type is taken, and they collectively form the personality type of a person. These combinations are—ISTJ, ISFJ, INFJ, INTJ, ISTP, ISFP, INFP, INTP, ESTP, ESFP, ENFP, ENTP, ESTJ, ESFJ, ENFJ, and ENTJ. Logistic regression, Naive Bayes, support vector machines, decision trees, and random forests are examples of classic machine learning techniques that have been effectively applied to predict MBTI personality types. The algorithm is explained with the help of a flow chart in Fig. 2.

1.3 Logistic Regression

In order to use logistic regression, first of all, sklearn.linear_model library is used and called. From that library, logistic regression is imported. A variable called model_log is created and in that variable logistic regression is called along with some parameters like max iterations set to 3000. C parameter or regularization parameter is called as well to reduce overfitting and increasing the number of parameters. n_jobs is used to specify the no. of cores to be executed the given code, and here, it is assigned the value -1 so that it stays on a single process and uses full power of CPU. This variable is then fit command to apply logistic regression with training datasets (X_test_tv,y_train) as parameters.

Now, the prediction variable y_pred is created to store the predicted value of the testing dataset. In order to print accuracy and difference between tested and actual data, confusion matrix, accuracy libraries are imported from sklearn.metrics. The

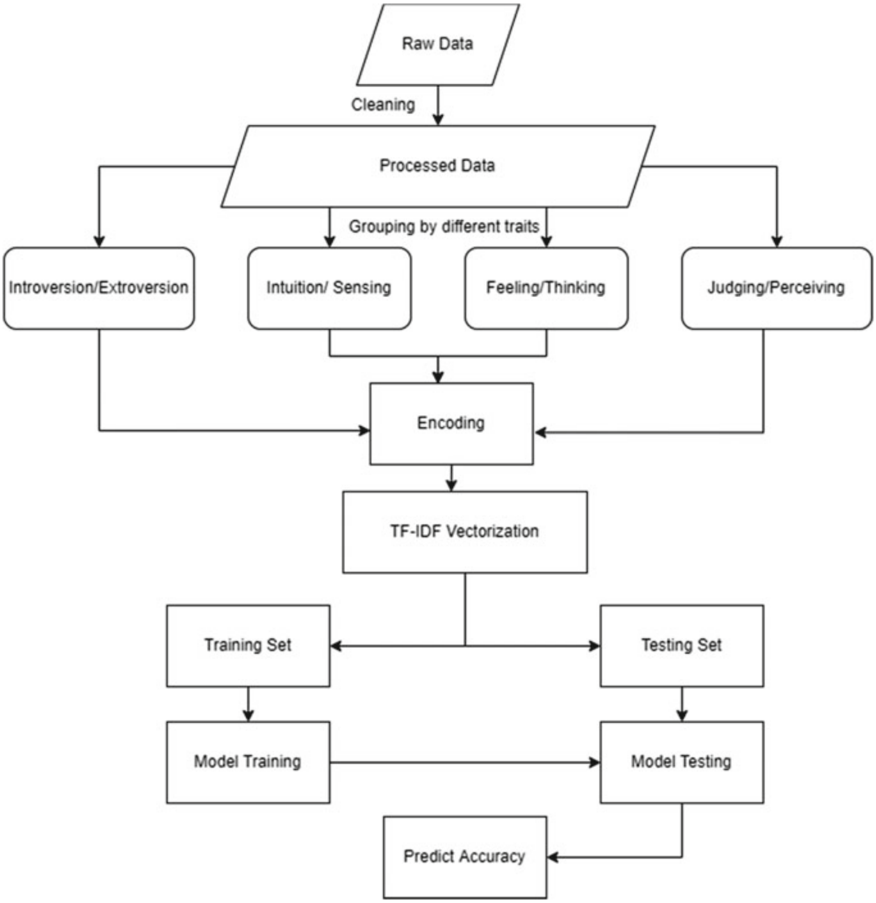


Fig. 2 System flow chart

confusion matrix is printed to highlight the difference followed by the accuracy score of the model.

The following is the equation of logistic regression

$$l = \frac{p}{1-p} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \frac{p}{1-p} = b^{\beta_0+\beta_1 x_1+\beta_2 x_2} \tag{1}$$

$$p = \frac{b^{\beta_0+\beta_1 x_1+\beta_2 x_2}}{b^{\beta_0+\beta_1 x_1+\beta_2 x_2}+1} = \frac{1}{1+b^{-\beta_0-\beta_1 x_1-\beta_2 x_2}} = s_b(\beta_0 + \beta_1 x_1 + \beta_2 x_2) \tag{2}$$

1.4 Support Vector Classifier

From SVM in scikit learn library, support vector classifier (SVC) is imported. A variable called model_SVC is created to store support vector classification function. This variable is then fitted with SVC model with training data (X_train_tv,y_train) as parameter followed by the creation of confusion matrix and returning the accuracy score.

$$J(w,\ b,\ a) = \frac{1}{2}w^T w - \sum\nolimits_{i=1}^{N} (a_i d_i)\left(\mathrm{w}^{\mathrm{T}} x_i + b\right) + \sum\nolimits_{i=1}^{N} a_i \tag{3}$$

At the optimum $\frac{\partial j}{\partial \omega} = 0$ and $\frac{\partial j}{\partial b} = 0\frac{\partial j}{\partial b} = 0$

$$\omega_0 = \sum_{i=1}^{N} a_i d_i x_i \text{ and } \sum_{i=1}^{N} a_i d_i = 0 a_i\left[d_i\left(\omega_0^T x_i + b_0\right) - 1\right] = 0 \tag{4}$$

either $a_i = 0$ or $d_i\left(\omega_0^T x_i + b_0\right) = 0$

$$J(\omega, b, a) = \sum_{i=1}^{N} a_i + \frac{1}{2}w^T w - w^T \sum_{i=1}^{N} a_i d_i x_{\mathrm{i}} - b\sum_{i=1}^{N}(a_i d_i) \tag{5}$$

$$Q(a) = \sum_{i=1}^{N} a_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{J=1}^{N} a_i a_J d_i d_J x_i^T x_J \tag{6}$$

1.5 Gaussian Naïve Bayes

Gaussian Naïve Bayes is used to find the best hypothesis(h) for given data(d), and the equation to find hypothesis is.

$$\mathrm{P(o|s) = (P(s|o) * \ P(o))/P(s)} \tag{7}$$

where.

- **P(o|s)** is the probability of output(o) when sample(s) has already happened
- **P(s|o)** is the probability of sample(s)when hypothesis has already happened
- **P(o)** is the probability of output(o) being true
- **P(s)** is sample(s) probability

$$\mathrm{mean(x) = 1/n * sum(x)} \tag{8}$$

$$\text{standard deviation}(z) = \text{sqrt}(1/n * \text{sum}(zi - \text{mean}(z)^2)) \tag{9}$$

This model is available on sklearn.naive_bayes as GaussianNB. model_gaussian_nb is the variable allocated for it, and then, this variable is fitted with training data as parameters. The confusion matrix and accuracy score are thus then printed.

1.6 Multinomial Naïve Bayes

$$\text{Posterior Probability} = \begin{array}{l}(\text{Conditional Probability} * \text{Prior Probability})\\ /\text{Predictor Prior Probability}\end{array} \tag{10}$$

$$P\left(\frac{A}{B}\right) = \left(\frac{P(A \cap B)}{P(B)}\right) = \frac{P(A) * P\left(\frac{B}{A}\right)}{P(B)}$$

where

P(A)	the probability of occurring A.
P(B|A)	the conditional probability of B occurring when A has already occurred.
P(A|B)	the conditional probability of A occurring when B has already occurred.
P(B)	the probability of occurring B.

This model is available sklearn.naive_bayes as MultinominalNB. model_multinomial_nb is the variable allocated for it, and then, this variable is fitted with training data as parameters. The confusion matrix and accuracy score are thus then printed.

1.7 Decision Trees and Random Forests

A decision tree is a classification technique that employs supervised learning. It is a tree like structure that contains a root node, a parent node, branches, and leaf node. In order to use this model, you have to import decision tree classifier from sklearn.tree module.

Just like a forest is made up of large number of trees, similarly random forests create decisions trees on data samples from each tree and choose the most efficient solution by voting. This model is available in the sklearn.ensemble library as random forest classifier.

2 Methodology for Personality Type Prediction

2.1 Dataset for Training the Model

The MBTI dataset or Myers-Briggs personality type dataset [21] is a dataset that has been taken from Kaggle Website. This is 60 Mb file which is in CSV format. This dataset consists of 8675 rows and 2 columns. The first column consists the information about the types of people belonging to one of the 16 personalities as mentioned, while the second column has their answers to the particular questions as well as their YouTube comments link, thereby communicating with them.

2.2 Proportionality in Dataset

The Seaborn, Python 2D visual library and Matplotlib, the Python 2D editing library, amongst other tools were used to preview the data and identify the MBTI personality types' distribution in the database. Figure 3 plots the number of instances of each MBTI personality type in the database using the Plotly library. It can be inferred from the pie chart that the number of INFP, INFJ, INTP, INTJ personality types accounts for more than 65% of total person. On the other hand, 10 personality types (ESTJ, ESFJ, ESFP, ESTP, ISFJ, ENFJ, ISTJ, ENTJ, ISFP, ISTP) out of 16 account for about 18% only.

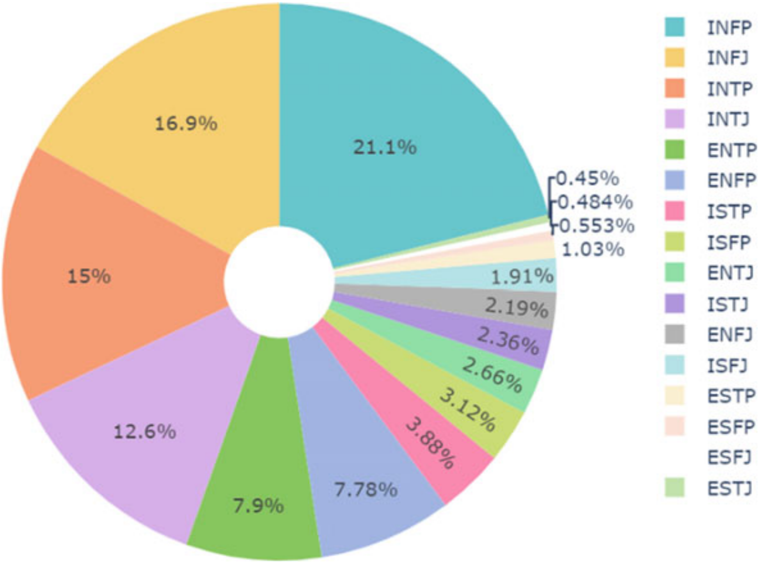


Fig. 3 Percentage of each MBTI personality type in the dataset

2.3 Four-Dimensional Categorization of Type Indicators

In order to categorize people, their personality is divided into 4 groups [17]. The first one decides either between introvert(I) or extrovert(E), whilst the second one decides either between intuitive(N) or sensing(S). The third group decides either between thinking(T) or feeling(F), and the last fourth group decides either between judgemental(J) or perceivers(P). Out of all these 4 groups, only one entry can be chosen from each other. Hence, there will be overall 16 different groups of people with unique personalities. Figure 4 displays segregated type indicator and it is observed that there is higher difference in count of I/E and N/S type, whereas there is little difference in case of count of F/T and J/P personality indicators.

From Table 1 and Fig. 4, we can clearly see that in the first phase of personality index, i.e. introversion/ extroversion, extroverts(E) are much more populated than their counterparts introverts(I). Similarly, in the second phase, which is intuition/sensing, the distribution of sensing people(S) is much higher than intuitive people(N). Figure 6 and Table 3 also show that in the third case of thinking/feeling, the distribution of thinking people(T) is slightly higher than feeling people(F). Finally,

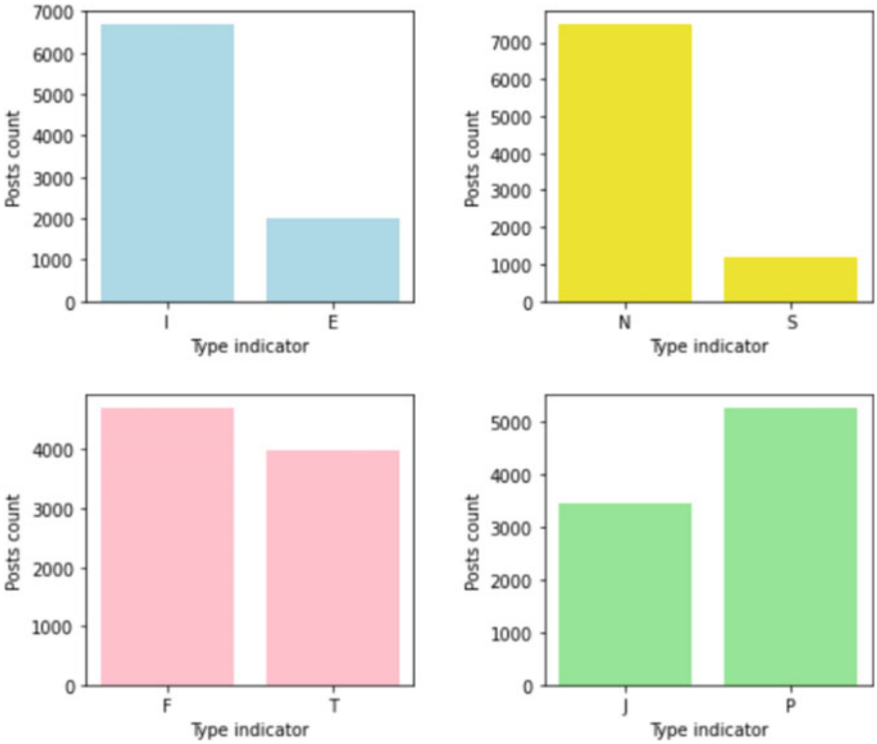


Fig. 4 No. of posts for different personalities type indicators

Table 1 Distribution across type indicators

Type indicator	Distribution
(I)ntroverts	1998
(E)xtroverts	6667
I(N)tuitive	1196
(S)ensing	7487
(T)hinking	4966
(F)eeling	3985
(J)udgemental	5247
(P)erceiver	3443

in the fourth category judging/perceiving, the population of people who are judgemental(J) according to their index is greater than of perceivers(P), thus pointing out that different features are not distributed equally amongst the phases.

2.4 Pre-processing the Dataset

Some trends were also noticed from the raw data, like the average number of words per post for each personality type, the number of "http" links which redirect to the respective posts, and both trends implied a higher online activity amongst introverts than extroverts. After that, some word removal was necessary as to make the data more precise and readable by the model, since the data is from the Myers-Briggs personality type dataset on the internet, as discussed earlier. Another important reason to clean the data was the occurrences of strings and phrases that were not very influential to the trends, repetitive, or just not required. As a result, from the NLTK library and corpus module, "stopwords" were imported to remove general unrequired words from sentences. Also, "re" was imported to substitute not required text with a space. "WordNetLemmatizer" was imported from the NLTK library to lemmatize words into their root form. After this step, the number of words per post for each type were analyzed again in the final (post-cleaned) data and compared with the original (pre-cleaned) data with their frequency. Figure 5 shows the change in the percentage of words after pre-processing the words, and the percentage shows the reduction in the number of words.

2.5 Segregating Different Type Indicators

In order to compare results with the existing projects, the approach used is to segregate data into various type indicators, and finally merging the results to predict accuracy. "Introvert" type indicator is separated from "extrovert". "Intuition" type indicator is

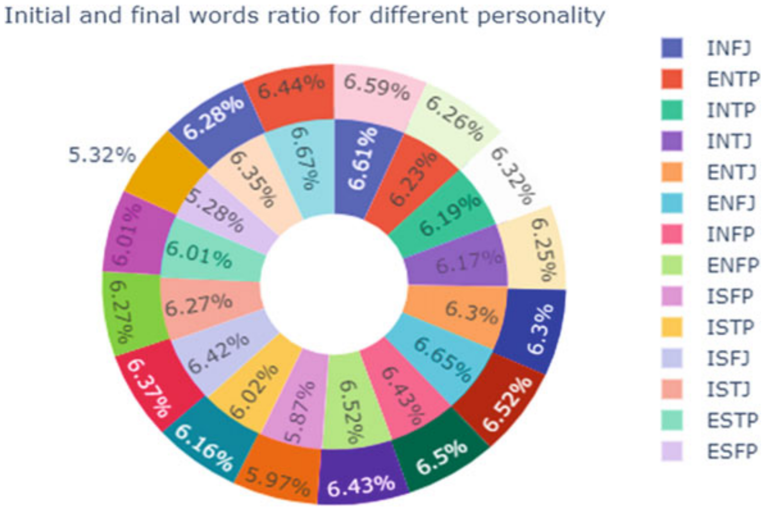


Fig. 5 Before and after processing word ratio of different personality types

separated from “sensing”. “Feeling” type indicator is separated from “thinking”, and “judging” type indicator is separated from “perceiving”. Encoding is on the newly formed rows to make it a binary classification problem.

2.6 Vectorize with Count and Term Frequency–Inverse Document Frequency (TF–IDF)

The TF-IDF Vectorizer from the Sklearn library was used to convert the words in the posts to a matrix or vector with 5000 maximum features in terms of relevance based on count. First of all, words from posts were converted to a vector of token numbers. In the next step, different models were applied and it returned a term-document matrix after learning the vocabulary dictionary. In training and testing set, “personality” columns were numbered using label encoder.

2.7 Classification Task

For predicting the personality type of a new person with entirely different posts, a classification model needs to be trained on a large number of data to learn from it. The models used in training were logistic regression, Naïve Bayes models—Gaussian and multinomial, support vector machine, decision trees, and random forests. These models were first applied to the dataset which was not segregated into type indicators and predicted the personality types in one go. Then, the same models were used on the dataset which was segregated into different personality type indicators and gave a single accuracy for each type indicator which will be then used to predict the overall accuracy. Grid search was used in models to get the parameters which helps in better accuracies.

2.8 Calculating Accuracy

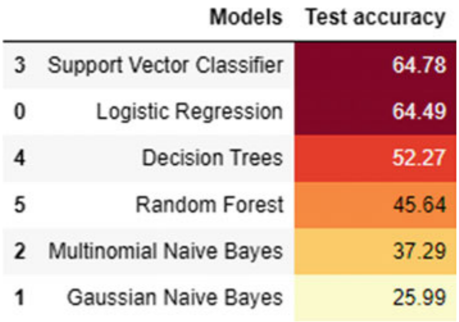
For the unsegregated dataset, there is one and only accuracy and needs no further changes, but for the dataset that was segregated into different parameters, and one model gave four accuracies one for each personality type indicator viz. I/E, N/S, F/T, J/P, the overall accuracy needs to get calculated. A function named "result" was created to calculate the overall score, which compares each predicted type indicator with the real-type indicator and adds 0.25 when both match and 0 when they do not. Since there are 4 different type indicator segregation viz. I/E, N/S, F/T, J/P, each have a weightage of 0.25, so when all the predictors are correct, the value is returned as 1, and if three are right and one is wrong, the result is 0.75.

Individual score (I/E or N/S or F/T or J/P) = score + 0.25 if individual prediction=individual test

Total score = (I/E score + N/S score + F/T score + J/P score)/total number of observations.

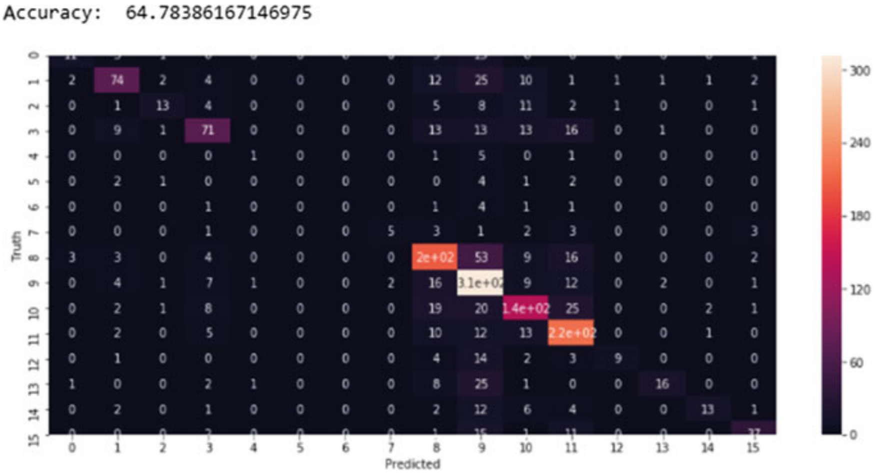
3 Results and Discussion

A dataset having many posts and their personality type was used to learn and then predict the personality of a completely unknown person using powerful NLP tools. There was more introverts' data than extroverts', similarly, more intuitive personality type data was available than sensing type, which was hindering from better learning because of low data availability. The model was created with the help of the NLTK module. The useful words were vectorized and using a machine learning model was trained and is now ready to test, confirmed by "wordcloud" to see the most frequent words for each word. The personality prediction was tokenized and lemmatized by NLTK libraries and vectorized by TF-IDF Vectorizer, on which machine learning models viz. logistic regression, support vector classifier, Gaussian Naïve Bayes, multinomial Naïve Bayes, decision trees, and random forest were applied. Support vector classifier gave the best result on the test set, which was about similar to logistic regression at 64% in unsegregated data as shown in Fig. 6. and 77% in segregated data as shown in Fig. 8. Figure 7 display confusion matrix of SVM classifier on unsegregated data; Figs. 9, 10, 11, 12 display various confusion matrices using SVM on segregated type indicators. Though the accuracy is higher in segregated data because of the new logical score calculating method and easier classification, this



	Models	Test accuracy
3	Support Vector Classifier	64.78
0	Logistic Regression	64.49
4	Decision Trees	52.27
5	Random Forest	45.64
2	Multinomial Naive Bayes	37.29
1	Gaussian Naive Bayes	25.99

Fig. 6 Accuracies on unsegregated data



```
Accuracy:  64.78386167146975
```

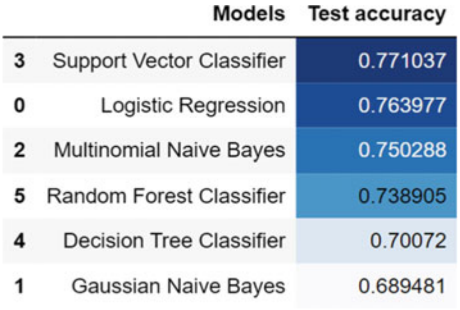
Truth \ Predicted	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
1	2	74	2	4	0	0	0	0	12	25	10	1	1	1	1	2
2	0	1	13	4	0	0	0	0	5	8	11	2	1	0	0	1
3	0	9	1	71	0	0	0	0	13	13	13	16	0	1	0	0
4	0	0	0	0	1	0	0	0	1	5	0	1	0	0	0	0
5	0	2	1	0	0	0	0	0	0	4	1	2	0	0	0	0
6	0	0	0	1	0	0	0	0	1	4	1	1	0	0	0	0
7	0	0	0	1	0	0	0	5	3	1	2	3	0	0	0	3
8	3	3	0	4	0	0	0	0	2e+02	53	9	16	0	0	0	2
9	0	4	1	7	1	0	0	2	16	3.1e+02	9	12	0	2	0	1
10	0	2	1	8	0	0	0	0	19	20	1.4e+02	25	0	0	2	1
11	0	2	0	5	0	0	0	0	10	12	13	2.2e+02	0	0	1	0
12	0	1	0	0	0	0	0	0	4	14	2	3	9	0	0	0
13	1	0	0	2	1	0	0	0	8	25	1	0	0	16	0	0
14	0	2	0	1	0	0	0	0	2	12	6	4	0	0	13	1
15	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Fig. 7 Confusion matrix on unsegregated data (SVM)

approach is more time as well as storage consuming. So, there is a trade-off between data, time, and accuracy in these approaches.

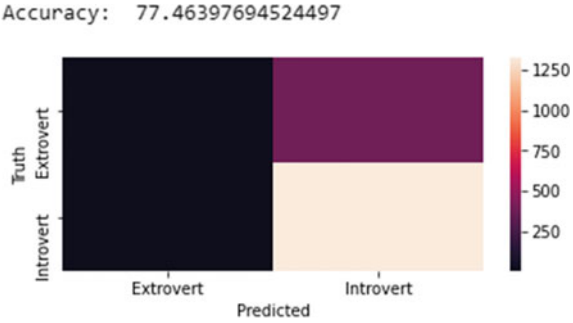
4 Conclusions

The main idea behind this project is to differentiate between the types of personality and to gain more understanding between everyone. Personality prediction has many real-time applications in the world, as it can be used to get career counselling, employment testing, relationship counselling. NLP can be used as an extension to messages to know the person's personality whilst texting and suggesting other activities he/she may like judging from their personality. NLP, when combined with social



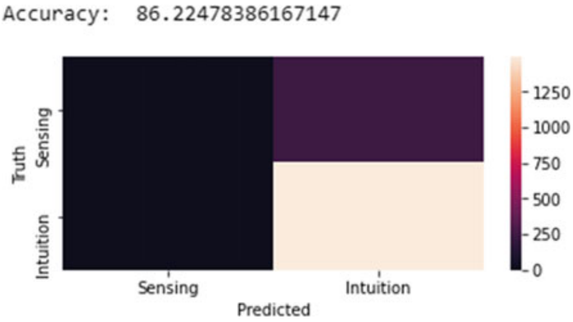
	Models	Test accuracy
3	Support Vector Classifier	0.771037
0	Logistic Regression	0.763977
2	Multinomial Naive Bayes	0.750288
5	Random Forest Classifier	0.738905
4	Decision Tree Classifier	0.70072
1	Gaussian Naive Bayes	0.689481

Fig. 8 Accuracies of different models on segregated data (SVM)



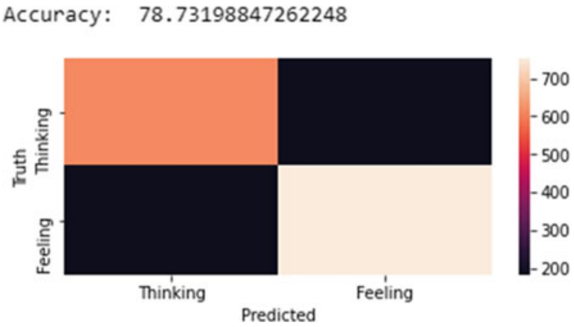
```
Accuracy:  77.46397694524497
```

Fig. 9 Confusion matrix I/E (SVM)



```
Accuracy:  86.22478386167147
```

Fig. 10 Confusion matrix N/S (SVM)



```
Accuracy:  78.73198847262248
```

Fig. 11 Confusion matrix F/T (SVM)

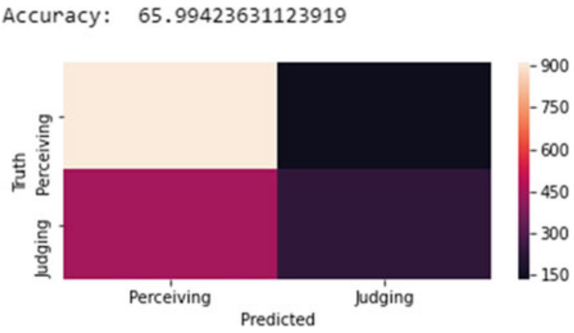


Fig. 12 Confusion matrix J/P (SVM)

media platforms, can be useful in advertising in targeting a specific personality type for better returns. For the training, we used the MBTI dataset from Kaggle, which was about 40 Mb of storage data. By using 6940 rows of data to train, we predicted about 1700 unknown rows with 64.78% accuracy in unsegregated data and 77% in segregated data using NLP to tokenize the words and calculate accuracy with new calculating method in case of segregated data. In both circumstances, the SVM has the maximum accuracy since it seeks to find the optimal margin to divide classes and lowers the possibility of data mistake. Segregated data is easier to classify as it has only two variables in each to distinguish, whereas in unsegregated data there are 16(different personality types), but in this case, the time complexity increases with increasing accuracy. TF-IDF approach is used in NLP to tokenize and predict personality and the accuracy can further be enhanced by using complex models like bidirectional encoder representations from transformers (BERT), Word2Vec, XLNet and using larger dataset for future aspects and for finding a better middle ground between time and accuracy. Also, artificial intelligence can learn better by personality detection models, to get more idea about feelings of humans and will become more understanding.

References

1. Charvet SR (1997) Words that change minds: mastering the language of influence. Author's Choice Publishing
2. Myers IB, McCaulley MH, Quenk NL, Hammer AL (1998) MBTI manual: a guide to the development and use of the Myers-Briggs Type Indicator. Consulting Psychologists Press
3. Davis PK, Bigelow JH (2001) Meta models to aid planning of intelligent machines. RAND GRADUATE SCHOOL SANTA MONICA CA.
4. Lima AC, de Castro LN (2013) Multi-label semi-supervised classification applied to personality prediction in Tweets. In: 2013 BRICS congress on computational intelligence and 11th Brazilian congress on computational intelligence, pp 195–203. IEEE
5. Hall LM, Bodenhamer BG (1997) Figuring out people: design engineering with meta-programs: deepening understanding of people for better rapport, relationships, and influence. JA KUBU

6. Wan D, Zhang C, Wu M, An Z (2014) Personality prediction based on all characters of user social media information. In: Chinese national conference on social media processing, pp 220–230. Springer, Berlin, Heidelberg
7. Sharma K, Kaur A (2015) Personality prediction of Twitter users with logistic regression classifier learned using stochastic gradient descent. IOSR J Comput Eng 17(4):39–47
8. Sultana M, Paul PP, Gavrilova M (2015) Social behavioral biometrics: an emerging trend. Int J Pattern Recognit Artif Intell 29(08):1556013
9. Ngatirin NR, Zainol Z, Yoong TLC (2016) A comparative study of different classifiers for automatic personality prediction. In: 2016 6th ieee international conference on control system, computing and engineering (ICCSCE), pp 435–440. IEEE
10. Aydin B, Kindiroglu AA, Aran O, Akarun L (2016) Automatic personality prediction from audiovisual data using random forest regression. In: 2016 23rd international conference on pattern recognition (ICPR), pp 37–42. IEEE
11. Amirhosseini MH, Kazemian HB, Ouazzane K, Chandler C (2018) Natural language processing approach to NLP meta model automation. In: 2018 international joint conference on neural networks (IJCNN), pp 1–8. IEEE
12. Gjurković M, Šnajder J (2018) Reddit: a gold mine for personality prediction. In: Proceedings of the second workshop on computational modeling of people's opinions, personality, and emotions in social media, pp 87–97
13. Bassignana E, Nissim M, Patti V (2020) Personal-ity: a novel youtube-based corpus for personality prediction in Italian. arXiv preprint arXiv:2011.05688
14. Al Marouf A, Hasan MK, Mahmud H (2020) Comparative analysis of feature selection algorithms for computational personality prediction from social media. IEEE Trans Comput Social Syst 7(3):587–599
15. Abidin NHZ, Remli MA, Ali NM, Phon DNE, Yusoff N, Adli HK, Busalim AH (2020) Improving intelligent personality prediction using Myers-Briggs type indicator and random forest classifier. Int J Adv Comput Sci Appl
16. Genina A, Gawich M, Hegazy A (2020) An approach for sentiment analysis and personality prediction using Myers Briggs type indicator. In: International conference on advanced intelligent systems and informatics, pp 179–186. Springer, Cham
17. Amirhosseini MH, Kazemian H (2020) Machine learning approach to personality type prediction based on the myers–briggs type indicator®. Multimodal Technol Interact 4(1):9
18. Choong EJ, Varathan KD (2021) Predicting judging-perceiving of Myers-Briggs Type Indicator (MBTI) in online social forum. PeerJ 9:e11382
19. Nisha KA, Kulsum U, Rahman S, Hossain M, Chakraborty P, Choudhury T (2022) A comparative analysis of machine learning approaches in personality prediction using MBTI. In: Computational intelligence in pattern recognition, pp 13–23. Springer, Singapore
20. Kadambi P (2021) Exploring personality and online social engagement: an investigation of MBTI users on twitter. arXiv preprint arXiv:2109.06402
21. Myers-Briggs Type Indicator dataset [Online] Available: https://www.kaggle.com/datasnaek/mbti-type

Multi-Document Abstractive Summarization for PPT Generation

Rajeswari Sridhar, Shreyas Thirumalai, Nikhil Anu, and P. Jayadev

Abstract In this paper, we aim to generate abstractive summaries of extensive news articles and a scientific corpus and present them in the form of a PowerPoint presentation. We use a pre-trained model run on a dataset comprising Wikipedia articles. This model is then fine-tuned by running it on a variety of datasets made public by the authors—Multi-News, Multi Science. Additionally, we employ the use of the popular Document Understanding Conference (DUC) (NIST Documents Understanding Conference, 2002) dataset as well. Our experiments also involve training the datasets from the ground up. We also compare our results to several publicly available baselines. Finally, the summaries generated through these datasets are passed through a paragraph separating algorithm to generate a PowerPoint presentation.

Keywords Word embeddings · Transformers · Seq2Seq models · Abstractive summarization

1 Introduction

The exponential growth of textual information in the modern digital age has produced a need for extracting useful information from a large amount of similar data efficiently. As a result, when a user queries, he is presented with much more than anticipated. Automatic text summarization systems are a solution to this challenge, helping the user decide the usefulness of a document without perusing it in its entirety. In recent times, these systems have undergone rapid development to produce high-quality summaries.

Two types of summaries can be generated: *extractive* and *abstractive*. Extractive summaries are relatively simpler to implement as they are generated by replicating

R. Sridhar (✉) · S. Thirumalai · N. Anu · P. Jayadev
Department of Computer Science and Engineering, National Institute of Technology,
Tiruchirappalli, TN 620015, India
e-mail: srajeswari@nitt.edu

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_63

sentences from the source document(s). Abstractive summaries are closer to human-written summaries as they generate words and phrases absent in the source document. Due to the nature of abstractive summaries, they are more challenging. Factors such as poor readability, factual inconsistency, and lack of coherence could produce unreasonable summaries.

Multi-document summarization (MDS) is the task of generating summaries from a cluster of homogenous documents. The task is usually more complex than single-document summarization. An ideal MDS system presents information organized around critical areas to represent diverse views while shortening the source text.

This paper summarizes news articles, i.e., producing an abstractive summary from multiple news sources relating to the same event. A single news event comprises articles from various news organizations. Thus, it is vital to present the user with a comprehensive summary covering the relevant information from all the different articles. Our goal is to train a model that produces high-quality results for MDS of extensive news articles. We also employ a scientific corpus and compare the results. We train the model on different datasets such as Multi-News [8], Multi-XScience [9], and DUC-2002 [12]. We compare the results and choose the model that generates a summary with the highest recall-oriented understudy for gisting evaluation (ROUGE) scores. The summary generated is presented in the form of a PowerPoint Presentation. The challenge in this regard is effective text segmentation which aids in splitting the summary into coherent sections. These sections can then be inserted into individual slides in the form of bullet points.

The rest of the paper is organized as follows: Sect. 2 deals with recent works in summarization, Sect. 3 discusses our proposed model for presentation generation, Sect. 4 discusses the results, and finally, Sect. 5 concludes the work with a possible extension.

2 Related Work

Recent work in MDS systems focuses on neural models that have attempted to exploit the graph structure in text clusters or text classification tasks. Several new works on abstractive summarization adapted sequence-to-sequence models to MDS and have achieved significant results. Recent works on multi-document summarizations [1–3] are implementations of Seq2Seq models. These works demonstrated the powerful ability to represent sequences in text for abstractive summarization.

Our model augments the proposed architecture in [3], which is based on a Seq2Seq model with a transformer architecture that can hierarchically encode documents. The work reported in [3] was trained using WikiSum [10], which is a large-scale dataset adapted from Wikipedia, and its relatively larger size compared to Document Understanding Conferences (DUC) was stated as reasoning for its usage. Our idea is to train the model on different datasets, Multi-News [8], Multi-XScience [9], and DUC [12], and measure the differences in performance. Zhang et al. [4] inspired us to employ a pre-trained model and fine-tune it using the DUC [12] dataset.

Text segmentation is another task that is vital for PPT generation. Alemi and Ginsparg [5] demonstrates several schemes that can be used to split a text into coherent paragraphs. The method relies on word-word co-occurrence statistics encoding meaningful semantic and syntactic relationships. There are three different schemes used which are: (1) *greedy splitting*: which at each step inserts the best available segmentation boundary; (2) *dynamic programming-based segmentation*: which uses dynamic programming to find the optimal segmentation; (3) *iterative refinement scheme*: which starts with the greedy segmentation and then adjusts the boundaries to improve performance.

3 Methodology

Figure 1 shows the block diagram of our methodology. Our first task is to prepare the abstractive MDS model. We measure the performance of various models using the famous metric ROUGE [7]. We compare the performances of (i) the base Hiersumm model [3] trained on the WikiSum dataset [10]; (ii) the model trained on DUC alone; (iii) the model trained on Multi-XScience [9] data alone; (iv) the pre-trained WikiSum model and fine-tune it using the DUC [12] and Multi-News [8] datasets. The goal here is to obtain the best ROUGE scores, implying good quality summaries, while summarizing extensive newspaper articles such as NewYork Times (NYT) and Wall Street Journal (WSJ).

As given in Fig. 1, the input multi-documents are made to run through a Seq2Seq model to generate a summary. This summary is segmented using a Word2Vec model to generate bullet point summary and organized as presentation slides. The methodology is expressed in detail in the following subsections.

3.1 Summary Generation

Hierarchical transformers model for summarization (Hiersumm) has been implemented recently. Firstly, we train the Hiersumm model from scratch on the DUC [12] dataset and after that use the pre-trained model made available by Liu et al. [3] and fine-tune it using the DUC dataset. Secondly, we fine-tune the model using Multi-News [8]. The approach works particularly well as these datasets are similar in vocabulary and complexity of the phrases. On the other hand, the approach did not provide good results with the Multi-XScience [9] dataset due to scientific jargon not well captured by the vocabulary and, consequently, the pre-trained hierarchical transformer model provided by Liu et al. [3]. We present our findings in the subsequent section. The summaries generated in this stage are passed to the PPT generation phase.

Datasets. In this work, we used the following datasets for training the models:

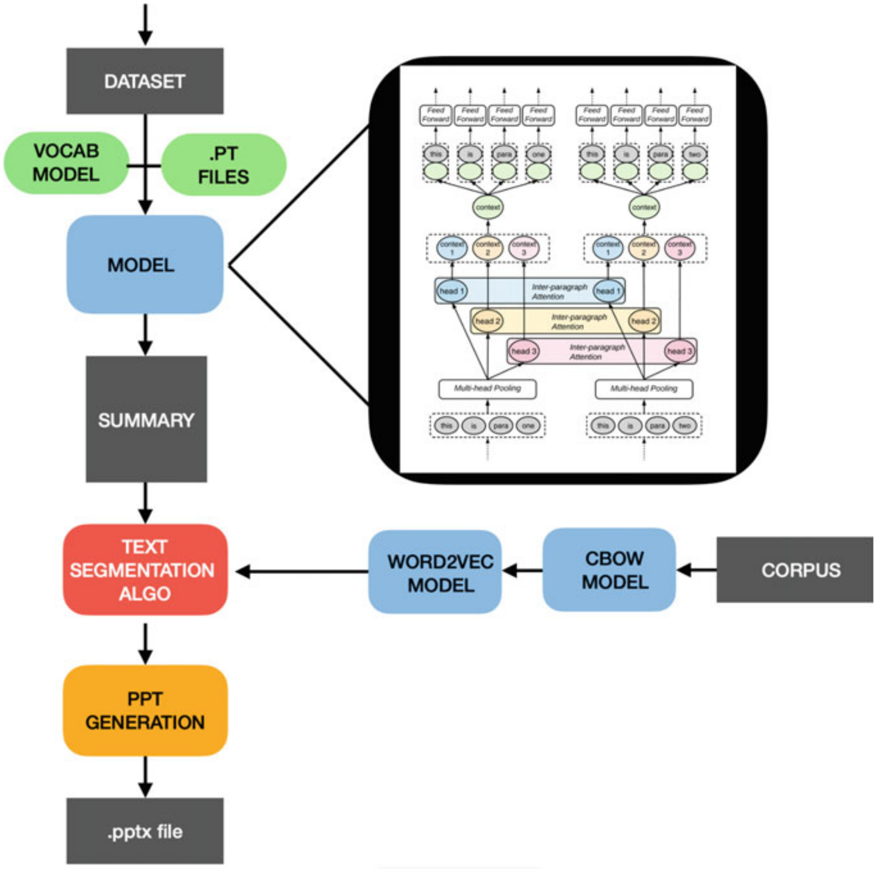


Fig. 1 Block diagram

DUC-2002 [12]. The DUC dataset is a popular dataset comprising (source, summary) pairs for both single and MDS tasks. Our project uses the 200-word summaries provided by the dataset.

Multi-News [8]. A large-scale MDS dataset that consists of articles and summaries collated from the site *https:// newser.com.* A professional human editor wrote every summary of the dataset.

Multi-XScience [9]. *A MDS dataset created using scientific articles.* The abstract and the citations of a paper form the multiple source documents of the summarization task, and the related work is the summary.

Data Preparation. A sentence piece [11] vocabulary file is required to prepare and feed the dataset into the Hiersumm model [3]. Byte-pair encoding is employed for this purpose. The sentence piece vocabulary file contains word-index mappings that encode the words before training and decode them during the translation phase. While the sentence piece model employed to train the WikiSum dataset presented in

[3] is publicly available, we choose to generate our vocab file for training. However, in one of our experiments, we use the vocabulary file to generate the requisite files to perform the training.

Dataset Generation. Using the vocabulary file, we encode the source and target in the format required by the Hiersumm model [3]. A sample is presented below.

```
# Multi Science
def MultiSciencePreprocess():
    batches ← Split data into batches
    for batch in batches:
        combined_src ← Combine abstract and references.
        encoded_src ← encode combined source
        encoded_target ← encode target text
    save result as.pt file ← Format required by pytorch
# Multi News
def MultiNewsPreprocess():
    # Dataset uses a special tag to separate stories for an example
    batches ← Split data into batches
    for batch in batches:
        examples ← split by newline character
        for example in examples:
            stories_list ← split by story separator tag
            stories_encoded ← encode using sentencepiece
            target_encoded ← encode summary
        save result as.pt file ← Format required by pytorch

    # GPU doesn't accept strings with encoded source text length > 5000
    if src text encoded length > 5000:
        split into chunks of 5000
# DUC
def DUCPreprocess():
    sets ← each sets contains a summary and multiple documents.
    for set in sets:
        docs_list ← Combine all the docs.
        encoded_src ← encode combined documents.
        encoded_target ← encode summary.
    save result as.pt file ← Format required by pytorch.
```

Abstractive Summarizer. The following are the four stages of the summarizer:

Encoder. The encoder used is a transformer encoder heavily inspired by the OpenNMT [6] implementation. It consists of several transformer layers as given in the algorithm that take in positional embedding of the source text. As input passes through the source text, the transformer layers feed it into the optimization layer.

```
#Encoder
class TransformerEncoder:
    num_layers ← number of encoder layers
    d_size ← size of the model
    heads ← number of heads
    d_ff ← size of the inner feed forward layer
    embeddings ← embeddings to use. Must have positional encodings.
```

```
        positional_ff_activation_fn ← position wise feed
forward layer's activation function

 def forward(source):
     out ← transpose embeddings and make contiguous
     mask ← invert sequence mask
     out ← run forward pass for all layers of transformer
     return embeddings, out
```

Decoder. The decoder passes the encoded text through a series of transformer decoder layers. The decoder layer is also required to supervise alignment guiding. An alignment is a common approach for word sense disambiguation that ensures the right place appears in the proper context. Finally, the decoder layer also makes use of an average attention network.

```
#Decoder
class TransformerDecoder:
    num_layers ← number of decoder layers
    d_size ← size of the model
    heads ← number of heads
    d_ff ← size of the inner feed forward layer
    copy_attention ← required for using separate copy attention
    self_attention_type ← type of self attention: average or
                                  scaled-dot
    dropout ← dropout in residual, self attention, feed forward
                    (prevent overfitting)
    embeddings ← embeddings to use. Must have positional
                         encodings
    maximum_relative_pos ← max distance between i/p in
                                  relative position representation
    aan_useffn ← turn on feed forward layer in average
                       attention network
    full_context_alignment ← enable full context decoder
                                  forward for alignment
    alignment_layer ← layer to supervise for alignment
                             guiding
    alignment_heads ← cross attention heads for alignment
                             guiding
def forward(target):
    for layer in transformer_layers:
        output, attention, attention_alignment ← pass through a
                                         transformer decoder layer
    output ← apply layer normalisation
    decoder_out ← transpose output
    return decoder_out
```

Translator. The translator takes the encoder–decoder model, the data, and the vocabulary and uses it to generate the final predicted summaries. This set of processes is done in batches with selected beam size and batch size for a required number of steps. The translations are stored and used to report rouge scores using the predicted text, the golden target text (reference), and raw source text.

```
#Translator
class Translator:
    model ← model used
    vocab ← vocab needed for translation
    n_best ← size of generated summaries
    beam_size ← beam size to determine number of search
                  candidates for the decoder
    start_token,end_token ← start and end tokens
    max_length, min_length ← maximum and minimum
                              sequence length

    def translate(data_iter,step):
        for batch in data_iter:
            batch_data ← fast translate each batch on the model.
            for b in range(batch_size):
                pred ← predicted sentences by building target
                        tokens from batch_data
                gold ← gold target sentences
                src ← decoded source sentences
                process symbols in the pred, gold, src
                translations ← append (pred,gold,src)
        write translations to file
        report rouge scores of the translations for each step
```

Optimizer. Optimizers are algorithms or methods used to change the attributes of your neural network, such as weights and learning rate, to reduce the losses. The optimizers that are used will define the procedure to change the weights or learning rates of the neural network to reduce losses. A learning rate of 0.05 is used in this work. Optimization algorithms or strategies are responsible for reducing the losses and providing the most accurate results possible. Here, the ADAM optimization algorithm is used.

```
#Optimizer
class Optimizer(object):
    method (obj) ← choose optimization algo
    lr ← learning rate
    lr_decay ← learning rate decay multiplier.
    start_decay_steps ← step to start learning rate decay.
    beta1, beta2 ← parameters for adam.
    decay_method ← custom decay options.

def build_optimizer(model, checkpoint):
    if (instance of Optimizer does not exist):
        instantiate the Optimizer class with empty
        optimizer state and parameters
    else:
        load the optimizer from a checkpoint
        update the model parameters based on current gradients
    return optimizer
```

Metrics. The following two metrics are used to evaluate the summaries:

Rouge. ROUGE stands for recall-oriented understudy for gisting evaluation. It is a popular metric that includes measures to determine the quality of a summary by comparing it to ideal human-written summaries. The measures count the number of overlapping units such as n-gram, word sequences, and word pairs between the computer-generated summary to be evaluated and the ideal summaries created by humans. ROUGE-N is an n-gram recall between a candidate summary and a set of reference summaries. It measures the informativeness of the generated summary and can be calculated as follows

$$P_{\text{ROUGE - }n} = \frac{\sum_{\text{gram}_n \in \text{IHS}} \text{count}_{\text{match}}(\text{gram}_n)}{\sum_{\text{gram}_n \in \text{CGS}} \text{Count}(\text{gram}_n)} \tag{1}$$

$$R_{\text{ROUGE}-n} = \frac{\sum_{\text{gram}_n \in IHS} \text{count}_{\text{match}}(\text{gram}_n)}{\sum_{\text{gram}_n \in IHS} \text{Count}(\text{gram}_n)} \tag{2}$$

$$F_{\text{ROUGE}-n} = \frac{2 \times P_{\text{ROUGE}-n} \times R_{\text{ROUGE}-n}}{P_{\text{ROUGE}-n} + R_{\text{ROUGE}-n}} \tag{3}$$

where IHS is the ideal human summary for reference, CGS is the computer-generated summary, n represents the length of the n-gram, gram $_n$, and Count$_{\text{match}}$ (gram $_{n}$) is the maximum number of n-grams co-occurring in a generated summary CGS and a human summary IHS. For our experiments, we use uni-gram and bi-gram for n and present the ROUGE-1 and ROUGE-2 scores. ROUGE-L score, in contrast, measures overlap between longest common subsequences (LCS), which can be a measure of readability. ROUGE-L score can be computed as follows:

$$P_{\text{ROUGE - }l} = \frac{\text{LCS (CGS, IHS)}}{|\text{ CGS }|} \tag{4}$$

$$R_{\text{ROUGE}-l} = \frac{\text{LCS (CGS, IHS)}}{|\text{ IHS }|} \tag{5}$$

$$F_{\text{ROUGE}-l} = \frac{2 \times P_{\text{ROUGE}-l} \times R_{\text{ROUGE}-l}}{P_{\text{ROUGE}-l} + R_{\text{ROUGE}-l}} \tag{6}$$

Bleu. BLEU stands for bilingual evaluation understudy and is another widely used technique to compare the translation of a text to a reference text. While ROUGE scores measure how much the n-grams from the reference summaries are present in the generated summaries, the BLEU score measures how many n-grams from the generated summaries appear in the reference summaries. In this way, they are complementary to each other. BLEU score is computed as follows:

$$\text{BLEU} = \text{BP.exp}\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{7}$$

where P_n is the geometric average of modified n-gram precisions up to length N and non-negative weights w_n totaling up to one. BP is the brevity penalty computed as

$$\mathrm{BP} = \begin{cases} 1, c > r \\ e^{(1-r/c)}, c \leq r \end{cases} \tag{8}$$

where c is the length of the generated translation, and r is the reference corpus length.

3.2 PPT Generation

The first step in this is the segmentation of text based on the coherence of sentences which aids in dividing the information into different slides. Word embedding is a learned representation for text where words with the same meaning have an equal representation. Word embedding is in fact a class of techniques where individual words are represented as real-valued vectors in a predefined vector space. Each word is mapped to one vector, the vector values are learned in a way that resembles a neural network, and hence, the technique is often lumped into the field of deep learning. Each word is represented by a real-valued vector, often tens or hundreds of dimensions which is contrasted to the thousands or millions of dimensions for sparse word representations, such as a one-hot encoding. The distributed representation is learned based on the usage of words which allows words that are used in similar ways to result in having similar representations, naturally capturing their meaning. Word embedding methods learn a real-valued vector representation for a predefined fixed-sized vocabulary from a corpus of text.

Word2Vec [15] is a statistical method for efficiently learning a standalone word embedding from a text corpus. Two different learning models were introduced that can be used as part of the Word2Vec approach to learning the word embedding continuous bag-of-words (CBOW) or continuous skip-gram model. The CBOW model learns the embedding by predicting the current word based on its context. Both models are focused on learning about words given their local usage context, where a window of neighboring words defines the context. This window is a configurable parameter of the model. The key benefit of the approach is that high-quality word embeddings can be learned efficiently (low space and time complexity), allowing more significant embeddings to be learned (more dimensions) from a much larger corpora of text (billions of words).

Alemi and Ginsparg [5] presents three segmentation algorithms. These are the *greedy*, *dynamic programming*, and *iterative refinement* approaches. The Word2Vec model is used to divide the summary into segments based on a segmentation algorithm. These segments are then inserted into each slide.

```
#PPT Generation
def TextSegmentation(summary):
    word2vec.bin ← word embedding model file generated using CBOW.
    return segmentation_algo (summary, word2vec) ← segment the summary

def CreatePPT():
```

Table 1 ROUGE scores

Training steps	F-Rouge-1	F-Rouge-2	F-Rouge-L
DUC	25.24/24.88	1.70/1.70	12.06/11.97
WikiSum + DUC	34.63/35.04	9.53/9.54	17.43/17.62
Multi-News	23.34/16.00	5.74/4.02	13.10/8.94
Multi-XScience	18.43/17.50	2.28/2.17	12.87/12.37

```
segments = TextSegmentation(summary) ← 2D array generated
for segment in segments:
  insert segment into new slide ← split it into bullet points
save as.pptx file ← Final ppt generated.
```

4 Experiments and Results

Our experiments use the PyTorch[1] framework, with code heavily inspired by the OpenNMT [6] implementation. Due to hardware constraints, the number of source tokens and target blocks were limited to 5 while training on the DUC [12] dataset.

4.1 Experiments

The training steps were restricted to 10 while training on the Multi-News dataset [8]. We modified the Hiersumm [3] model to load checkpoints every 10 steps while balancing memory constraints. The Multi-XScience [9] dataset posed some difficulties as well. We were constrained to using a batch size of 3000 that did not give good results. [3] recommends using a batch size of 10,000 or greater, which supersedes our memory limitations. Finally, 30,000 source documents exceeded the maximum length constraints while training on Multi-News [8]. We split the documents into parts of the maximum length (5000). While this did not impact the encoding and decoding aspects of the training process, it might have impacted identifying the hierarchical relationships between the documents as these parts were elements of the same document. We cannot provide proof of the same due to hardware constraints for the comparison. The following is a tabulation of our results based on the calculations using Eqs. (6)–(8) (Tables 1 and 2).

[1] https://pytorch.org.

Table 2 BLEU scores

Training steps	B-1	B-2	B-L
DUC	26.3	6.6	3.6
WikiSum + DUC	26.2	6.4	2.9
Multi-News	49.7	11.4	2.3
Multi-XScience	26.8	5.4	0.6

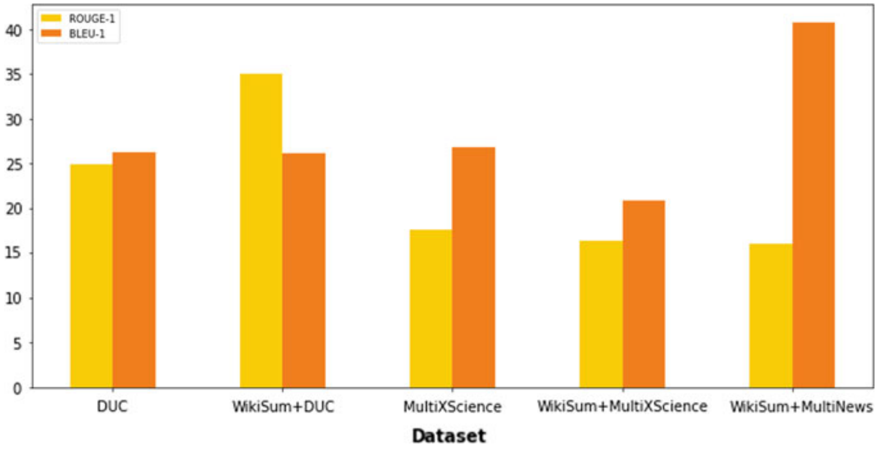


Fig. 2 Rouge versus bleu

4.2 Results

Thus, it is evident that the Wiki + DUC model fared much better than the others models as evident from Fig. 2. It has the highest ROUGE scores indicating the quality of the summaries generated. Interestingly, the ROUGE scores see a significant improvement with fine-tuning the pre-trained Hiersumm [3] model with 5000 steps using the DUC-2002 dataset. Using these results, we generate the PPT. The results inferred that the greedy and DP approach produces similar segmentation of sentences which could be attributed to the small size of the data. These methods provide slightly contrasting results for more extensive texts, but for our use case, the summaries are between less than 200 words and hence provide similar results. Thus, the scheme chosen is inconsequential for our use case.

4.3 Constraints

A significant obstacle faced in applying end-to-end models to MDS is the sheer size and number of source documents. As a result, it was practically infeasible given our hardware memory limitations to training a model that encodes them into vectors and generates a summary. In our case, we were constrained to using the GPU Google Colab allotted us. In particular, our experiments were not entirely successful on the Multi-XScience dataset owing to the usage of a batch size that was lesser than the one recommended in [3]. Thus, this experiment has scope for producing better results with better resources. Lu et al. [9] demonstrated promising results with Multi-XScience on Hiersumm [9], albeit using the same vocabulary file made public by its author. On the other hand, we found a vocabulary file trained from the ground up with limited resources provided better results.

5 Conclusion and Future Work

Thus, this paper demonstrates the PPT generation of news articles using MDS. While our implementation was limited to news articles and a scientific corpus, this could be adapted to various other domains such as legal texts. In particular, the generation of a PPT could be helpful for executive summaries in a corporate setting wherein brevity is of utmost importance.

References

1. Wang Q, Ren J (2021) Summary-aware attention for social media short text abstractive summarization. Neuro-Computing 425:290–299
2. Liang Z, Du J, Shao Y, Ji H (2021) Gated Graph Neural Attention Networks for abstractive summarization. Neurocomputing 431:128–136
3. Liu Y, Lapata M (2019) Hierarchical transformers for multi-document summarization. Artif Intell ACL 2019, pp 5070–5081
4. Zhang J, Tan J, Wan X (2018) Adapting neural single-document summarization model for abstractive multi- document summarization: a pilot study. Assoc Comput Linguist, pp 381–390
5. Alemi AA, Ginsparg P (2015) Text Segmentation based on semantic word embeddings, Inf Retrieval, KDD, pp 1–10
6. Klein G, Kim Y, Deng Y, Senellart J, Rush AM (2017) OpenNMT: open-source toolkit for neural machine translation. Artif Intell, pp 67–72
7. Lin CY (2004) ROUGE: a package for automatic evaluation of summaries. Assoc Comput Linguist, pp 74–81
8. Fabbri A, Li I, She T, Li S, Radev D (2019) Multi-news : a large-scale multi-document summarization dataset and abstractive hierarchical model. In: Proceedings of the 57th annual meeting of the association for computational linguistics, pp 1074–1084
9. Lu Y, Dong Y, Charlin L (2020) Multi-XScience: a large-scale dataset for extreme multi-document summarization of scientific articles. EMNLP 2020, pp 8068–8074
10. Liu PJ, Saleh M, Pot E, Goodrich B, Sepassi R, Kaiser L, Shazeer N (2018) Generating wikipedia by summarizing long sequences. Comput Lang ICLR 2018 (p.arXiv preprint arXiv:1801.10198)
11. Kudo T, Richardson J (2018) SentencePiece: a simple and language independent subword tokenizer and detokenizer for Neural Text Processing, EMNLP 2018, pp 66–71
12. NIST documents understanding conference 2002

13. Ba JL, Kiros JR, Hinton GE (2016) Layer normalization (p.arXiv preprint arXiv:1607.06450)
14. Allahyari M, Pouriyeh S, Assefi M, Safaei S, Trippe E, Gutierrez J, Kochut K (2017) Text summarization techniques: a brief survey. Int J Adv Comput Sci Appl IJACSA 2017 8:397–405
15. Meyer D (2016) How exactly does word2vec work? Uoregon. Edu, Brocade. Com, pp 1–18

DIMDA: Deep Learning and Image-Based Malware Detection for Android

Vikas Sihag, Surya Prakash, Gaurav Choudhary, Nicola Dragoni, and Ilsun You

Abstract With the widespread adoption of handheld smartphones, the number of malware targeting them has grown dramatically. Because of the widespread use of cell phones, the quantity of malware has grown dramatically. Because of their ubiquity, android smartphones are the most sought-after targets among smart gadgets. We provide an unique image-based deep learning system for android malware detection in this article. The suggested system predicts if an application is malicious or genuine based on network traffic represented in picture format. The proposed method is tested against 13,533 applications from various banking, gambling, and utilities industries. Our technique is effective, with an accuracy of 98.44% and a recall of 98.30%. It also outperformed conventional machine learning methods.

Keywords Malware analysis · Android · Deep learning · Network traffic

V. Sihag (✉) · S. Prakash
Security and Criminal Justice, Sardar Patel University of Police, Jodhpur, India
e-mail: vikas.sihag@policeuniversity.ac.in

S. Prakash
e-mail: spu19cssp@policeuniversity.ac.in

G. Choudhary · N. Dragoni
DTU Compute, Department of Applied Mathematics and Computer Science, Technical University of Denmark (DTU), Lyngby, Denmark
e-mail: gauravchoudhary7777@gmail.com

N. Dragoni
e-mail: ndra@dtu.dk

I. You
Department of ICT Environmental Health System, Soonchunhyang University, Asan, South Korea
e-mail: ilsunu@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_64

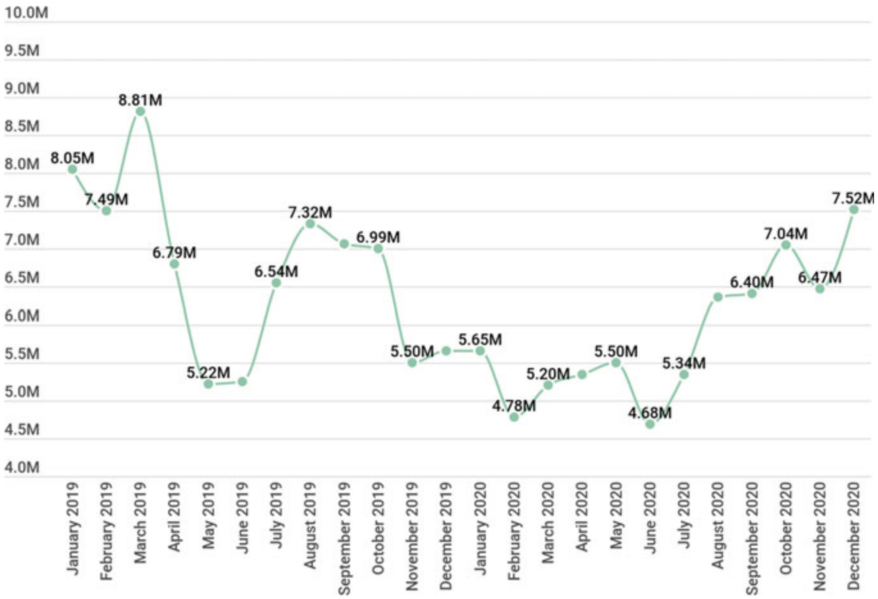


Fig. 1 Average monthly attacks on mobile users [6]

1 Introduction

In comparison to other operating systems, Google android is now the most popular operating system on the planet, with 80% of customers using it. Recently, the exponential sale of over one billion android smartphones has been seen, and the company now ranks first in the smartphone industry with a market share of more than 85% [17]. 65 billion app downloads from Google Play android devices have become pervasive in today's technological world, leading to the new threat of a new malware set in the android system. The advanced techniques are not enough by only detecting malicious android apps [9, 12, 23, 25]. Nowadays, Covid, a popular term, is used to hide malware, spyware, viruses, and Trojan droppers like covid.apk, covid Mapv.apk, anti covid.apk, covid Mappia.apk, and covid detect.apk. These applications were installed on fraudulent Web sites, and URLs were disseminated via spam. Figure 1 depicts the rise in monthly attacks on smartphone users.

The rapid growth of malware targeting android devices has demanded the development of effective localization methods for detecting a zero-day threat. Android OS allows user to download and install apps from third party marketplaces, which lack the requisite skills or equipment to check for infection in submitted applications. Security operation of android is most probably relies on permission-based mechanism permissions are nothing, but the certain request made by the applications that they are required to access the interface data [22].

Table 1 Threat agents involvement in cyberthreats

Threats	Threat agents						
	Corporations	Cyber criminals	Cyber terrorists	Hacktivists	Insiders	Nation states	Script kiddies
Botnets	●	●	●	○		●	○
Cyber espionage	○				○	○	
Data breaches	●	●	●	●	●	●	○
Denial of service	○	●	○	●		○	●
Exploit kits	●	●				●	
Identity theft	●	●	○	●	●	●	○
Information leakage	●	●	○	○	○	●	○
Insider threat	●	●	○			○	
Malware	●	●	○	○	○	●	○
Phishing	●	●		●	●	●	○
Physical damage	●	●	○	○	●	●	○
Ransomware	●	●			○	●	○
Spam	○	○			●	○	
Web application attacks	●	●	○	●		●	○
Web-based attacks	●	●	●	●		●	○

●: Primary threat group

○: Secondary threat group

Google Play, for instance, uses the Bouncer tool to validate uploaded apps. In any case, it has been recently proven that the Bouncer dynamic examination cycle may be avoided by employing a few uncomplicated emulation opponents. The main problem with the emulator is that some malware can hide, and the emulator cannot give us the precise result we want to achieve in the detection phase of an actual device. Furthermore, traditional server-side-based malware detection cannot be ignored [8]. On Google's Play Store, you may get a variety of antivirus programs such as AVG, Norton, and Avira, which commonly employ pattern-matching techniques for malware recognition [3]. Analysis of top cyberthreats with reference to threat agents is presented in Table 1. Threats are being differentiated in the table based on being primarily or secondarily being deployed by threat agents.

This paper introduces a security framework based on deep learning and images called as DIMDA (Deep learning and image-based malware detection in android) to mitigate malicious in android platform. We start by generating dynamic analysis logs for a specific android executable file in form of network capture, then transform network traffic into sessions and display each session as an image. Finally, generated images are used to train a deep learning model. In addition, the model was tested against the dataset to see if it could detect malware.

Organization of the paper: The following section discusses related work. Section 4 presents the proposed framework, followed by findings and analysis in Sect. 5. Finally, Sect. 6 brings the work to a conclusion by outlining future directions.

2 Related Work

Investigating android malware samples is time-consuming and error-prone [19, 20]. Multiple approaches have been presented in literature to counter malwares with static, dynamic, or hybrid analysis methods. Yu [31] focused on automatic techniques to work suspicious application screening, and the antivirus enterprise significantly wants threat-level evaluation. The authors discussed the Droid Screening framework, an efficient way to accelerate the android malware investigation; it is how classification algorithms classify models by learning malicious evidence features. Another approach is discussed by Odusami et al. [16]. The authors discussed existing methods what detection the approaches by which they can achieve accuracy detection techniques to use static dynamic and machine learning approaches for better results by using different machine learning approaches; higher accuracy can be achieved. The identification of network anomalies is a vital and rapidly evolving research field in which Monowar et al. [4] discussed different types of attacks in adios. Profile organizing is a faster way to differentiate the data from the machine learning process. The existing malware types and their malicious behavior are classified in Table 2.

Recent works, Kouliaridis et al. [11] focused on android malware detection approaches in different types of classification and how the signature-based anomaly-based and hybrid-based detection techniques are applicable. This paper is the amalgamation of various analyses of anomalies created in the market and how packages and multiple corresponding analysis (MCA) is done. Juniper Networks' Global Threat Centers discovered a 400-fold rise in android malware during summer 2010 [21]. Some well-known instances from Iker Burguera and Urko Zurutuza are "Fake Player," "Geinimi," "PJApps," and "HongToutou." Burguera et al. [5] address the crowd-drawing approach and provided a robust platform for crowdsourcing the collecting of overall evidence from an infinite number of actual individuals. The design behavior-based technique enriches system calls, and particular behavior apps are discussed. There is also a discussion about sensitive malware, which have traces from 50 traces from the application ten traces of malicious ones. Using several machine learning classifiers in combination with android malware detection offers potential advantages besides improved accuracy. Other methods to use the varied capabilities of the constituent classifiers include supplementing white-box assessment with careful supervision of intermediate output from far more intelligible base models.

Suleiman et al. [30] proposed an approach for the evaluation of the malware samples, which includes the investigation for the proactive android malware detection example of droid dream. These were affected 5000–20,000 users reported by Syman-

Table 2 Malware types and their behavior

Malware type	Definition	Malicious behavior							
		AU	CV	CC	DU	SD	Sp	SI	CR
Adware	Automatically displays or downloads advertising material	T							
Bot	Infects a computer to carry out commands under the remote control		T	T		T			
Cryptominer	Uses victim's computaional resources to mine cryptocurrencies								T
Ransomware	Locks the system or user files for ransom					T			
RAT	Allow an attacker to remotely control an infected computer		T	T	T			T	
Scareware	Scares people into visiting spoofed or infected websites				T				
Spyware	Enters a user's computer, gathers data from the device and sends it to third parties							T	
Virus	Copy itself and infect a computer without permission						T		
Worm	Self-replicating application which performs malicious activities					T	T		

[AU: Annoying user, CV: Creating vulnerabilities, CC: Command and control command execution, DU: Deceiving user, SD: Service denial, Sp: Spreading, SI: Stealing information, CR: Computational resource consumption]

Table 3 The most recent comparative research on android malware detection methods and learning methodology

Year	Authors	Analysis	Learning	Detection	Features
2014	Yuan et al.	Static & dynamic	Supervised	Binary	Behavior
2016	Alzaylaee et al.	Static	Supervised	Classification	Behavior
2017	Nix et al.	Static	Supervised	Overview	System calls
2018	Odusami et al.	Hybrid		Classification	Signature, permissions
2018	Belal et al.	Static	Supervised	Detection	Network traffic
2019	Burguera et al.	Dynamic	ML	Detection	System calls
2020	Chen et al.	Dynamic	ML	Detection	Spatial features
2020	Kouliaridis et al.	Dynamic	ML	Detection	Footprints
2021	Sihag et al.	Dynamic	DL	Binary	System & binder calls
2021	Proposed	Dynamic	DL	Binary	Network traffic

tec. In this paper, there is an overview of what different types of API calls-related Nike get m contact receiver trip to extract various features like command related which are metadata signatures permissions. The power-efficient method for detecting mobile threats in emergent communication networks is given by Yi et al. [7]. In this, portable engineers offer to program for nothing to collect client data for information examination and promoting purposes. A model could also be an unrestricted informal organization framework that allows clients to communicate with companions and to gather client addresses for advertisements. Table 3 shows the state of art comparison work on android malware detection approaches, and learning methodologies are used by different authors in the area with malware detection nowadays.

3 Static and Dynamic Analysis

Malware analysis analyzes a software application for different artifacts. During static analysis, an application is analyzed without executing it. Static analysis performs structural analysis of the sample. Static analysis tools can assist detect memory corruption problems and validate the accuracy of models for a specific system. Assisting tools are typically used for static analysis. It enables an analyst to grasp an application's logic. Android application is typically created using the Java programming language and the Android SDK. Java bytecode is converted to Dalvik executable (DEX) format after the build process. Static analysis, in other words, is an assessment of requirements, design, and code that differs from traditional dynamic testing in many key aspects. The major purpose of this investigation is to uncover flaws and determine whether or not they are likely to cause failures. Static analysis, like reviews, looks for defects rather than failures. Signature-based detection and heuristic-based detection are two ways that have been developed for it. These methods worked effectively against known harmful programs, but they were unable to detect new malware [15].

The dynamic analysis utilizes the executable's behavior and activities during execution to determine whether it is malware. The most important thing to remember about dynamic analysis systems is that they only run the binary for a short timeframe. Because malicious apps do not monitor their actions while only running for a few seconds, dynamic systems must show the binary execution over a considerable period of time. As a result, dynamic assessment is resource intensive in terms of hardware and time [28].

4 Proposed Framework

In the proposed framework, first, we predict the label of android APK network traffic of a real android device.

Overview of DIMDA Figure 2 presents the architecture of DIMDA. It aims to classify the sample application APK based on its network behavior as malicious or benign. The said application is executed in an emulator [18]. Network interactions of the application are captured and preprocessed to be represented in a vector form. A trained deep learning model uses the vector representation of network behavior for classification. The proposed approach consists of the below modules mentioned.

Dynamic Analysis In this module of the proposed framework, a dynamic analysis technique is applied to get app network interactions when executing on the underlying android operating system. An emulation sandbox is used to execute the app.

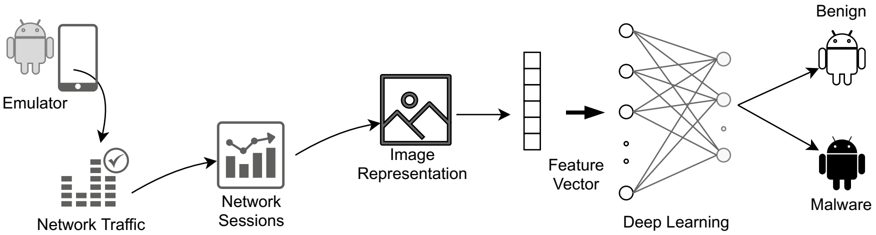


Fig. 2 Architecture of DIMDA

Copperdroid [24] emulator is configured and employed for app execution, and TCP dump is used for capturing network data.

Image Generation Network traffic captured from the above step is of different sizes, and traffic classification requires the adjacent units of traffic to be discrete ones. Our approach uses sessions to define network interactions. A session is the collection of flows in both directions corresponding to a connection. A flow is a set of packets having the same five identifiers: IP addresses both at the source and destination, port numbers at both the origin and destination, and protocol. Starting 784 bytes were considered. Steps involved for image generation are as follows:

i. Raw contiguous traffic is split into sessions.
ii. Empty sessions are ignored.
iii. Duplicate sessions are deleted.
iv. If session size more than 784 bytes, consider first 784 bytes.
v. If session size less than 784 bytes, padded with 0s.
vi. Trimmed session represented as 28 × 28 grayscale image.

Deep Learning We employ deep neural networks for learning and classification. The trimmed session represented as 28 × 28 grayscale images are fed into a multilayer NN model for malware detection. Multiple configurations were tested, and `128 × 64 × 1` layer configuration was considered. It is comprised of hidden layers with 128 and 64 neurons, with each layer densely connected. The output layer consisted of a single-neuron (sigmoid activation function), and the ReLU activation function was used for hidden layers. The model was fitted and trained for 64 batch sizes and 500 iterations. 10% random samples were selected for testing purposes.

Convolution neural networks (CNNs) are neural networks with multiple convolution layers (with each layer comprising multiple artificial neurons). They are primarily employed for classification and image processing. Neurons in neural network are function which takes in the weighted sum op inputs and gives functional output. An image, when passed into a CNN, passes through multiple layers developing several activation functions. For instance, first layer interprets major edges followed by interpretation of their combination and details in subsequet layers. Each layer of nodes learns from the output of the previous layer (feature set). As a result, nodes in each subsequent layer are able to identify increasingly complex, detailed components—picture visual representations.

Parameters of model:

- Layer—128
- Epoch—100
- Batch size—64
- Iteration—500
- Activation function—Relu.

Dataset and evaluation Performance and efficiency of `DIMDA` are evaluated and compared with other benchmark approaches. A dataset of android applications consisting of 13,533 samples was collected from different sources (benign samples =

2821 and malicious samples = 10,712) [10, 13, 14, 27]. The dataset comprised malware of different types and infection methods. To evaluate the DIMDA framework, parameters considered are as follows: true-positive rate (% of malware samples properly identified), true-negative rate (the proportion of benign samples properly identified), false-positive rate (% of samples mistakenly identified as malware), false-negative rate (% of improperly detected samples deemed harmless), precision, recall, F-measure, and Accuracy.

5 Experimental Results

DIMDA framework has experimented on Intel Core i7 2.4 GHz process with 12 GB RAM running Ubuntu 18.04 OS. CopperDroid emulator was configured to resemble real devices, as some malware tend to evade emulation environments by monitoring themselves. Of the dataset collected, we were able to execute 13,533 samples. Many samples were either not corrupted or were not running on the emulator. Multiple malicious and benign samples were also not performing any network activity.

Traffic captured from malware and benign samples during dynamic analysis was disintegrated into the session. Duplicate and empty sessions were deleted during image generation. Image generation gave us 219,490 image representations of sessions. 21,949 random images were selected for the test dataset. Rest images were fed into deep learning model for learning and fitting. Binary cross-entropy was used for loss function.

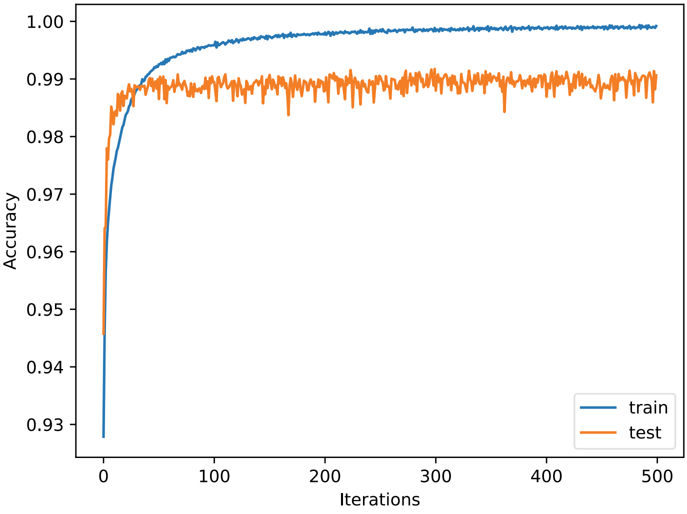


Fig. 3 Average accuracy of DIMDA versus iterations

Table 4 Comparison of DIMDA with existing deep learning solutions

Paper	Features	Detection (%)
Wang et al. [26]	Network traffic	97.8
Alzaylaee et al. [2]	API calls, Intents, Permissions	97.8
Xu et al. [29]	API, system calls, Network, Permissions	94.7
Sihag et al. [21]	System calls, binder calls	98.08
Abuthawabeh et al. [1]	Flow statistics	87.75
DIMDA	Network sessions	98.3

Figure 3 illustrates the accuracy of the DIMDA training and testing dataset over 500 iterations with 64 batch sizes. With the increase in iteration, accuracy increases. After 150 iterations, it is observed that the rise in accuracy is not steep. When evaluated using testing dataset, the results are as follows: TPR = 98.30%, TNR = 93.23%, FPR = 6.76%, FNR = 1.69%, precision = 98.44%, recall = 98.30%, accuracy = 97.37%, F-measure = 98.37%, and errorrate = 2.6%. As F-measure is considered to be relevant for unbalanced datasets, our approach suggests a promising F-measure of 98.37%. Comparison of DIMDA with other up to date deep learning approaches is shown in Table 4. SPSVERBc1framework can be enhanced in the future by taking into account various analyses, both dynamic and static characteristics (For instance, system calls, APIs, permissions, and network statistics data).

6 Conclusion and Future Work

The technique recommended incorporates an android malware detection framework. By running sample apps in an emulated environment, it extracts network traffic patterns for the real-world dataset. The session data from network traffic are preprocessed and displayed as grayscale images. For training and testing, these pictures are put into a deep learning model. The proposed method improves android malware detection. It exhibits better or comparable performance (accuracy 98.44 % and recall 98.30 %). DIMDA framework can be expanded in the future to add aspects other than network characteristics (for example, system calls, APIs, permissions, and network statistics information). For the detection of malware families, visual image analysis might be used.

References

1. Abuthawabeh M, Mahmoud K (2020) Enhanced android malware detection and family classification using conversation-level network traffic features. Int Arab J Inf Technol 17:607–614
2. Alzaylaee MK, Yerima SY, Sezer S (2020) Dl-droid: Deep learning based android malware detection using real devices. Comput Secur 89:101663
3. Bedford A, Garvin S, Desharnais J, Tawbi N, Ajakan H, Audet F, Lebel B (2016)Andrana: quick and accurate malware detection for android. In: International symposium on foundations and practice of security. Springer, pp 20–35
4. Bhuyan MH, Bhattacharyya DK, Kalita JK (2013) Network anomaly detection: methods, systems and tools. IEEE Commun Surv Tutor 16(1):303–336
5. Burguera I, Zurutuza U, Nadjm-Tehrani S (2011) Crowdroid: behavior-based malware detection system for android. In: Proceedings of the 1st ACM workshop on security and privacy in smartphones and mobile devices, pp 15–26
6. Chebyshev V (2021) Mobile malware evolution 2020. https://securelist.com/mobile-malware-evolution-2020/101029/
7. Chen CM, Liu YH, Cai ZX, Lai GH (2020) A power-efficient approach to detect mobile threats on the emergent network environment. IEEE Access 8:199840–199851
8. Feng R, Chen S, Xie X, Meng G, Lin SW, Liu Y (2020) A performance-sensitive malware detection system using deep learning on mobile devices. IEEE Trans Inf Forensics Secur 16:1563–1578
9. Johnson C, Khadka B, Basnet RB, Doleck T (2020) Towards detecting and classifying malicious urls using deep learning. J Wirel Mob Netw Ubiquitous Comput Dependable Appl 11(4):31–48
10. Kadir AFA, Stakhanova N, Ghorbani AA (2016) An empirical analysis of android banking malware. Protect Mobile Netw Dev Challenges Sol 209
11. Kouliaridis V, Barmpatsalou K, Kambourakis G, Chen S (2020) A survey on mobile malware detection techniques. IEICE Trans Inf Syst 103(2):204–211
12. La Marra A, Martinelli F, Mercaldo F, Saracino A, Sheikhalishahi M (2020) D-bridemaid: a distributed framework for collaborative and dynamic analysis of android malware. J Wirel Mob Netw Ubiquitous Comput Dependable Appl 11(3):1–28
13. Mahdavifar S, Kadir AFA, Fatemi R, Alhadidi D, Ghorbani AA (2020) Dynamic android malware category classification using semi-supervised deep learning. In: 2020 IEEE international conference on dependable, autonomic and secure computing, intl conf on pervasive intelligence and computing, intl conf on cloud and big data computing, intl conf on cyber science and technology congress (DASC/PiCom/CBDCom/CyberSciTech). IEEE, pp 515–522
14. Mobile C (2013) Mobile malware mini dump. [EB/OL]. [2016-6-12]. http://contagiominidump.blogspot.com
15. Nath HV, Mehtre BM (2014) Static malware analysis using machine learning methods. In: International conference on security in computer networks and distributed systems. Springer, pp 440–450
16. Odusami M, Abayomi-Alli O, Misra S, Shobayo O, Damasevicius R, Maskeliunas R (2018) Android malware detection: a survey. In: International conference on applied informatics. Springer, pp 255–266
17. Sihag V, Mitharwal A, Vardhan M, Singh P (2020) Opcode n-gram based malware classification in android. In: 2020 fourth world conference on smart trends in systems, security and sustainability (WorldS4). IEEE, pp 645–650
18. Sihag V, Swami A, Vardhan M, Singh P (2020) Signature based malicious behavior detection in android. In: International conference on computing science, communication and security. Springer, pp 251–262
19. Sihag V, Vardhan M, Singh P (2021) Blade: Robust malware detection against obfuscation in android. Forensic Sci Int Digital Invest 38:301176
20. Sihag V, Vardhan M, Singh P (2021) A survey of android application and malware hardening. Comput Sci Rev 39:100365

21. Sihag V, Vardhan M, Singh P, Choudhary G, Son S (2021) De-lady: deep learning based android malware detection using dynamic features. J Internet Serv Inf Sec (JISIS) 11(2):34–45
22. Sinha R, Sihag V, Choudhary G, Vardhan M, Singh P (2021) Forensic analysis of fitness applications on android. In: International symposium on mobile internet security. Springer, pp 222–235
23. Talegaon S, Krishnan R (2020) Administrative models for role based access control in android. J Internet Serv Inf Secur 10(3):31–46
24. Tam K, Khan SJ, Fattori A, Cavallaro L (2015) Copperdroid: automatic reconstruction of android malware behaviors. In: Ndss
25. Thang NC, Park M (2020) Detecting malicious middleboxes in service function chaining. J Internet Serv Inf Secur (JISIS) 10(2):82–90
26. Wang S, Chen Z, Yan Q, Yang B, Peng L, Jia Z (2019) A mobile malware detection method using behavior features in network traffic. J Netw Comput Appl 133:15–25
27. Wei F, Li Y, Roy S, Ou X, Zhou W (2017) Deep ground truth analysis of current android malware. In: International conference on detection of intrusions and malware, and vulnerability assessment. Springer, pp 252–276
28. Willems C, Holz T, Freiling F (2007) Toward automated dynamic malware analysis using cwsandbox. IEEE Sec Privacy 5(2):32–39
29. Xu L, Zhang D, Jayasena N, Cavazos J (2016) Hadm: hybrid analysis for detection of malware. In: Proceedings of SAI intelligent systems conference. Springer, pp 702–724
30. Yerima SY, Sezer S, Muttik I (2014) Android malware detection using parallel machine learning classifiers. In: 2014 Eighth international conference on next generation mobile apps, services and technologies. IEEE, pp 37–42
31. Yu J, Huang Q, Yian C (2016) Droidscreening: a practical framework for real-world android malware analysis. Sec Commun Netw 9(11):1435–1449

Personality Recognition for Candidate Screening

Piyush Patil, Saloni Goyal, Tanya Dwivedi, and Suvarna Bhat

Abstract In this paper, we perform personality recognition for the development of a platform for analysis and examination of emotions and behavior of job candidates through personality traits recognition. Personality traits can be considered an important factor for working in a professional environment. Evaluation of such traits at a preliminary stage can prove to be beneficial in a working medium. We have decided to explore textual inputs for developing an ensemble model that gathers the information from text responses, integrates them with a provided standard, and displays a clear and understandable way of assessing candidates' interest and enthusiasm. Hence, this research suggested an empirical technique to compare machine learning models such as support vector machine, Naive Bayes, decision tree, random forest, logistic regression, recurrent neural networks to discover the optimum personality recognition performance along with natural language processing (NLP), and affective computing methods and find better accuracy of classification algorithms than previous studies by merging big five dataset and MBTI dataset. Five human personality traits that are Extraversion (EXT), Neuroticism (NEU), Agreeableness (AGR), Conscientiousness (CON), and Openness (OPN) operated for problem analysis. The outcome revealed that support vector machine and logistic regression performed better overall than other models across all metrics with an average accuracy score 78.01% for EXT, 79.63% for AGR, 58.91% for NEU, 74.21% for CON, and 81.56% for OPN traits. However, the Naive Bayes algorithm resulted in overall lower performance.

P. Patil (✉) · S. Goyal · T. Dwivedi · S. Bhat
Computer Engineering Department, Vidyalankar Institute of Technology, Mumbai, India
e-mail: piyush.patil@vit.edu.in

S. Goyal
e-mail: saloni.goyal@vit.edu.in

T. Dwivedi
e-mail: tanya.dwivedi@vit.edu.in

S. Bhat
e-mail: suvarna.bhat@vit.edu.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_65

Keywords Personality traits recognition · Machine learning classification models · Natural language processing · Big five traits · Candidate screening

1 Introduction

Emotion recognition through text input is a demanding assignment that surpasses conventional sentiment analysis. Besides detecting basic responses such as neutral, positive, or negative, the objective is to pin down a set of emotions characterized by a higher gradient. Many subtleties are factored in to perform an accurate detection of human emotions, where context dependency is of prime importance.

There are different methods of tackling natural language processing problems, but they are majorly classified as rule-based and learning-based techniques. Rule-based approaches target more on pattern-identification and are largely based on grammar analysis and sentence structure, whereas learning-based approaches prioritize probabilistic modeling and likelihood maximization. This study prominently focuses on learning-based methodologies. In this research, we have chosen text mining to streamline unstructured data and to find out personality traits which depends on the "big five personality" model and MBTI dataset.

Emotion identification and human personality classification are two distinct disciplines of study, each with its own theoretical underpinnings. However, they have quite same learning-dependent methods. The main aim is to provide a broader assessment of the user's emotions, since it can get through the learning of a characteristic of the person, and personality traits analysis would provide a new point to understanding human emotion.

Psychology researchers mainly believe that there are five categories, or core factors, also called Big 5 that determines one's personality. To mention this model, the acronym OCEAN (for Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism) is generally used. Due to its popularity and clarity, we have chosen to use this precise model. Openness is a measure that describes an emotion, intellectual curiosity, willingness to try new things, general awareness, reaction to surroundings, etc. Conscientiousness is a scale that elaborates on how people control and regulate their senses and behavior. Extraversion indicates how people function and flourish in their general surroundings and react around others. Agreeableness reflects individual decisions in general concern for social harmony. Neuroticism measures negative emotions, mental stability, anger, anxiety, etc.

Judgment of personality is key when it comes to the prediction of characteristics of an individual which differentiates them from other individuals. Hence, getting a deeper knowledge of a person's personality types is extremely valuable for indicating their physical and mental well-being. Myers–Briggs Type Indicator (MBTI) is an indicator of an individual's perception of the world and decision-making process. It classifies an individual's personality based on Sensing(S) or Intuition(N), Thinking(T) or Feeling(F), Extraversion(E) or Introversion(I), and

Judging(J) or Perceiving(P). Information from these personality tests helps companies better comprehend the extent of their employees' strengths, weaknesses, and their capability to perceive and process information. As mentioned in Bajic [1], data obtained from MBTI assessments helps businesses build stronger organizations as it guides them to assemble efficient teams, facilitate communication between the team and the manager, motivate employees, solve conflicts, and develop leadership.

The main goal is to build a tool capable of recognizing the personality traits of a candidate, given a text containing his answers to pre-established personal questions with the support of statistical learning methods. Getting a general sense of a candidate's educational background, career history, and interest in the company should help you weed out unqualified candidates. Primary checks include looking at qualifications such as work experience, academic background, skills, knowledge base, traits, and behaviors that indicate a person's behavior, as well as competency. With traditional recruiting methods, recruiters struggle to evaluate candidates accurately. The purpose of our model is to determine if the candidate has the basic skills, aptitude, and enthusiasm to meet the requirements of the job. In candidate selection, various tools are used to assess a candidate's suitability for the job, including interviews, skills tests, psychometric tests, group discussions, and reference checks. It should be noted that the purpose of our model is to narrow down the most qualified candidates who should be treated to a traditional interview and remaining steps of candidate selection.

2 Related Works

Rahman et al. [2] present a great method to determine the way of detection of person's characteristics by making comparisons between several activation functions such as sigmoid and leaky ReLU. Python 2.7 is used for coding of the model, along with Google's word2vec embeddings and Mairesse features. The experimental analysis uses big five personality traits that are EXT, NEU, AGR, CON, and OPN. Calculating the median F1-score of the functions sigmoid, tanh, and leaky ReLU gives 33.11%, 47.25%, and 49.07%, respectively.

Pramodh and Vijayalata [3] Datasets stream-of-consciousness and MyPersonality are used. MyPersonality dataset has a mixture of 250 users who are updating around 10,000 Facebook Status Updates. Natural Language Toolkit has been used for their model, and their F1-scores are given as 0.665, 0.632, 0.625, 0.624, and 0.637 for the traits OPN, CON, EXT, AGR, and NEU, respectively.

Tareaf et al. [4] propose a model that can properly distinguish between a religious individual and a non-believer across 83% of circumstances, between individuals of Asian and European decent in 87% of situations, and between emotionally stable and emotionally unstable individual across 81% of circumstances. The presented analysis is a MyPersonality dataset containing 738,000 users who have granted their Facebook activities, data from other social networks, egocentric networks, and demographic characteristics.

Anatoli de Bradk e and ephane Reynal [5] explored different art models including text, audio, and video in multimodal emotion recognition.

Under text input, the dataset was a study by Pennebaker and King consisting of 2468 essays. Bag-of-words and word2vec embedding techniques were used for preprocessing. Personality scores were assessed by the Big Five Inventory. Different classification models were tested against each other such as multinomial Naive Bayes, support vector machines, recurrent neural networks, and LSTM.

In Majumder et al. [6], deep convolutional neural network or DCNN was used by the researchers to classify between personality traits on the basis of the big five model. The dataset used included the stream-of-consciousness essay dataset by James Pennebaker and Laura King. The model also utilizes Google's word2vec embeddings along with Mairesse features.

In Abyaa et al. [7], supervised learning algorithms have been used for the classification of personality according the big five model. The algorithms used were SVM, RF, k-nearest neighbors (kNN), Naive Bayes (NB), J48, logistic regression (LR), and bagging. The dataset used was collected from 48 students. Data included is basically of educational data, survey responses, EMA, and sensor data. Positives results of 62.5%, 50%, and 62.5% were achieved by Extraversion SVM, random forest, and logistic regression, respectively. However, only Naive Bayes and bagging give great results: 57.14% for Openness. SVM and bagging gave uplifting outcome of 50% for Agreeableness. In case of Neuroticism, only SVM gave acceptable results of 57.14%.

Tinwala and Rauniyar [8] proposed a model for personality detection using deep convolutional neural networks. The dataset used is essays collated by Pennebaker and King which are based on the big five model a.k.a the five-factor model or the OCEAN model. The document-level feature extraction was carried out using Google's word2vec embeddings and Mairesse features. The data processed is then fed to a deep convolutional network (DCN), and a binary classifier is used for classifying the availability or non-availability of the personality trait. Function of tanh is best used for traits Extraversion, Neuroticism, and Agreeableness giving F1-scores of 61.2%, 66.33%, and 62.67%, respectively. Sigmoid is best used for Openness and Conscientiousness providing F1-scores of 69.71% and 67.46%, respectively.

Kumar and Gavrilova [9] study proposes a personality traits classification system, which can incorporate the language-based features, that are formulated upon count-based vectorization and technique of Global Vectors word embedding, with a collection of predictive system that consists of decision trees and an support vector machine classifier. The given mixture of results helps to reliably find out the personality traits by the most recent tweets from the given profile. The proposed system's performance gets validated on a giant, publicly available Twitter MBTI Personality Dataset, and is compared favorably with other different state-of-the-art techniques.

3 Dataset

The dataset that we have chosen is from Kaggle [10] that is based on the Myers–Briggs Type Indicator Myers [11] and the NEO Personality Inventory Costa and McCrae [12] also called the Big 5. The Myers–Briggs Type Indicator (or MBTI for short) is a dataset of personality into 16 various types with four major divisions that are Introversion (I), Extraversion (E), Intuition (N), Sensing (S), Thinking (T), Feeling (F), Judging (J), and Perceiving (P). Depending on these attributes' personalities, it can be coded in a four-letter term for example—ISTJ, ISTP, and ESFJ. This dataset has more than 8500 rows of data, where each row is a MBTI type (four-letter MBTI code/type) of the person and a written entry of each of the last 50 things they have posted. The second dataset used is the MyPersonality project dataset (Kosinski et al. [13]) which consists of 2400 stream-of-consciousness texts labeled with a personality from Pennebaker and King [14] and used by Mairesse et al. [15] that examines five personality traits that are Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism. Major traits of both the personality indicators exhibit a correlation among them explained in Furnham [16] and hence can be used together on the dataset for increased accuracy. Judging–Perceiving dimensions exhibit similarity with Conscientiousness; Thinking–Feeling dimensions for the MBTI are equivalent to Agreeableness; Introversion–Extraversion correlates with Extraversion, and Openness correlates with the Sensing Intuitive MBTI type. Only Neuroticism does not correlate with any of the type in MBTI and somewhat inconsistent across all of them. The dimension remarkably missing from the MBTI is Neuroticism.

4 Methodology

The following is the proposed system's operation procedure: (i) merging dataset and re-sampling of the imbalanced date, (ii) preprocessing and text vectorization, (iii) personality classification based on text data using machine learning models, (iv) comparing the efficiency of the models with other classifiers, and (v) various evaluation metrics.

The publicly available dataset of big five personality traits is acquired from the website. The dataset is of 2467 rows, in which each row is shown as individual user. Each user's essays is included along with that user's big 5 personality type (e.g., Openness, Agreeableness). We merged this big 5 dataset with the MBTI dataset to increase the efficiency of the machine learning model. MBTI dataset is also publically available on Kaggle. This dataset consists of 8675 rows, where the user's past social media posts have been given with the MBTI personality type (e.g., ENTP, ISJF). We merged the data based on the correlation derived from the research paper of Adrian [16] (Furnham [16]). As a result, a labeled dataset comprises a total of 11,142 records.

4.1 Text Preprocessing

1. *Removal of Stop Words*

   Stop words, such as articles, prepositions, pronouns, conjunctions, and others, are the most common words in any language and do not provide any significant information to the text. "the", "a", "an", "so", and "what" are some of the stop words in English. Elimination of these stop words is the first step in text preprocessing. Stop words are present in abundance and hence can hamper the results while training the dataset. By keeping unwanted words out of our corpus, we can focus more on the high-level information essential for training our dataset.
2. *Handling imbalanced dataset*

   As shown in Figs. 1 and 2, the dataset is unevenly distributed in all five types, mentioned as follows: S/N Trait: S = 7466 and N = 1194, T/F Trait: T = 4685 and F = 3975, I/E Trait: I = 6664 and E = 1996, J/P Trait: J = 5231 and P = 3249. The outcome of any algorithm applied on a skewed or unbalanced classified dataset always favors the large and smaller classes that are passed over for prediction. So, we used re-sampling methods such as oversampling and undersampling which basically make the dataset equally distributed.

A. Undersampling

The minority classes can present information that is pertinent to the outcome, but the biased distribution of the classes can lead to ineffective results. Undersampling is the removal of random samples from the majority class. Undersampling minimizes the size of the data, requiring less time for learning. The drawback is that deleting majority classes may result in the majority class losing useful information.

B. Oversampling

Another strategy used to add minority cases to the dataset in order to achieve a balance is oversampling, which involves duplicating the current minority samples. This increases the data size but provides impetus to the minority classes.

4.2 Text Vectorization

1. *CountVectorizer*

   It is a technique for converting a given text to the vector or matrix. It counts the count of word and then converts it in vector format. It considers how many times a word appears in a text (multiplicity), ignoring grammatical subtleties and even word order. CountVectorizer creates the matrix which contains word with associated row for all the documents. The value of each cell is the number of words in that particular text sample. The frequency of a particular word in a text is proportional to the importance of the term in the text.
2. *Term frequency–inverse document frequency (TF-IDF)*

   TF-IDF technique is of two sections that are term frequency and inverse of document frequency. The term frequency is measured as a percentage of the total

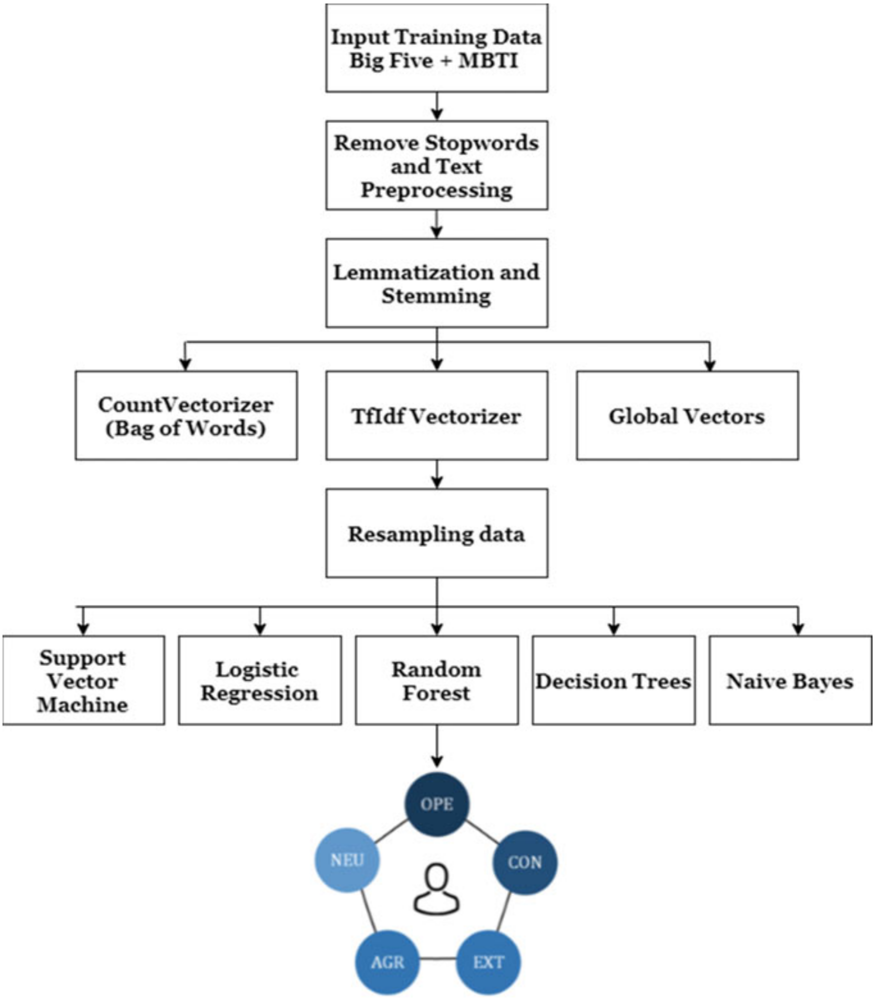
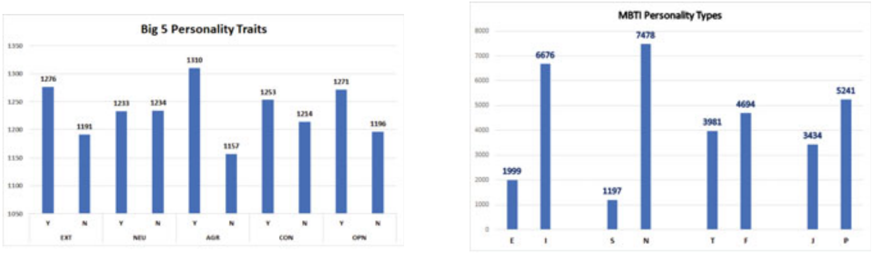


Fig. 1 Proposed text-based personality recognition framework

number of words in a single document. Term frequency does not consider the importance of words only the frequency. Some words can be most frequently present but are of little significance and hence can alter the results. Each word is given a weight based on its frequency in a corpus using inverse document frequency. This measure is generated by dividing the total number of documents divided by the number of documents which have that specific word.



(a) Occurrence of Big 5 Personality Traits

(b) Occurrence of MBTI Dataset Traits

Fig. 2 Imbalanced dataset

4.3 *Modeling*

4.3.1 Support Vector Machine

SVMs are popular linear classifiers that essentially split the data plotted on a multidimensional graph with a hyperplane. Ideally, the features we use in this classifier must correlate linearly with all the classes in a way that the classifier linearly segments the space to include all records with the same trait labels. For each label, we train a single-label one versus all classifier that employs the subset of features that are most linearly related to the classes they are assigned to. Finding the right hyper-parameter can be difficult, but it can be done by experimenting with various combinations and observing which ones work best. The method involves the creation of a grid of hyper-parameters and trying all of its various combinations and, hence, is called grid search. As mentioned in GeeksforGeeks [17], GridSearchCV is a built-in feature that uses a dictionary to specify the variables that can be used to train a model. The parameter grid is type of dictionary, having keys as parameters and settings are the values.

4.3.2 Decision Tree

The decision tree classifier builds the classification model by making trees for taking decisions. Every node of the tree is defined as a test, and every branch from that node is value as given in KDnuggets [18]. By learning simple decision rules inferred from past data, the purpose of employing a decision tree is to develop a training model that can be used to predict the class or value of the target variable (training data). When utilizing decision trees to forecast a record's class label, we start at the top of the tree. The record's attribute and the root attribute's values are compared. We jump to the next node depending on the comparison by the branch that to that value.

4.3.3 Naive Bayes

Naive Bayes works fairly well for text-based classification problems. The classifier makes predictions based on the learning of distribution of posterior probability. We train the Naive Bayes model with a subset of features listed in the previous section. It turns out that the Naive Bayes classifier yields the best result for one of the personality traits. This algorithm has a simple and intuitive design and is a good benchmark for classification purposes.

$$P\left(A/B\right) = \frac{P\left(B/A\right) * P(A)}{P(B)}$$

4.3.4 Logistic Regression

Logistic regression is one of the classification methods that has its roots in statistics but is now widely used in machine learning. It is a technique for calculating a data collection, where one or more variables influence the outcome. It is used to create the best-fitting model and characterize the relationship of the dependent variables and independent variables. In logistic regression Raj [19], the Sigmoid function is used in logistic regression to show probability values. In converts number to the range of 0 and 1.

4.3.5 Random Forest

Random forest is a decision tree-based model; it predicts the result by averaging several results from multiple pre-built decision trees. Each decision tree has different structures learned from the training input. In our project, multiple features are in use, but we have do not know how to weigh the features to get the best results for the classification. So we experimented with the random forest model with the training set containing the features we choose to use and use the rest of the labeled data for validation.

5 Results and Discussion

We processed with the big five dataset for personality traits classification for candidate screening. We initially used the CountVectorizer (bag of words) method for the vectorization the words on the five personality traits dataset and then trained using support vector machines, logistic regression, Naive Bayes algorithm, random forest, and decision trees models. In addition, we applied hyper-parameter tuning for support vector machines such as GridSearchCV to get better accuracy. We got

Table 1 Data essays: vectorizer–bag of words

	SVM	DT	NB	LogR	RF	SVM (Hyper-parameter tuning)
cEXT	55.46	51.81	50.41	55.66	53.84	55.26
cNEU	51.61	54.45	51.82	53.23	56.27	58.91
cAGR	52.83	50.41	50.61	53.41	54.04	54.04
cCON	50.81	51.41	53.03	53.03	56.27	53.44
cOPN	57.08	52.42	53.03	56.03	59.11	58.90

the better accuracy of NEU traits by hyper-parameter tuning. The comparison of the result is given below: (Table 1).

Then, we used the TF-IDF vectorizer that takes not only the frequency but also the importance of words into account when analyzing a corpus. With the TF-IDF vectorizer, we observed that accuracy increased for all the classification traits by around an increase of 2%. The results with recurrent neural networks are also not up to the mark.

Nevertheless, based on the big 5 dataset, we could not achieve high accuracy. Following the research of personality traits in humans and getting a sense of the concept from the paper of Furnham [16], we merged both five personality traits dataset and the MBTI dataset. By the help of CountVectorizer and TF-IDF vectorizer, we convert words into vectors. Then, we trained the data on different machine learning models. By merging the MBTI dataset with the big five personality, we achieved an increase of accuracy by around 20% (Tables 2, 3 and 4).

Table 2 Data essays: vectorizer–TfIdf

	SVM	DT	NB	LogR	RF	SVM (Hyper-parameter tuning)
cEXT	53.84	55.66	49.79	55.06	53.64	55.66
cNEU	58.09	49.59	52.83	57.89	56.47	55.87
cAGR	56.07	51.61	50.61	55.87	49.79	56.07
cCON	54.64	51.01	52.22	55.66	52.83	53.44
cOPN	60.72	53.23	53.03	60.52	62.14	60.92

Table 3 Data essays + MBTI: Vectorizer–CountVectorizer

	SVM	DT	NB	LogR	RF	SVM (Hyper-parameter tuning)
cEXT	72.31	72.45	69.71	75.91	71.42	78.86
cNEU	N/A	N/A	N/A	N/A	N/A	N/A
cAGR	70.39	68.28	54.82	72.11	71.01	75.81
cCON	68.05	64.61	57.37	69.62	59.62	72.22
cOPN	77.47	76.08	75.63	79.97	78.66	81.15

Table 4 Data essays + MBTI: Vectorizer–TfIdf

	SVM	DT	NB	LogR	RF	SVM (Hyper-parameter tuning)
cEXT	**78.01**	70.48	68.56	77.65	72.05	77.47
cNEU	N/A	N/A	N/A	N/A	N/A	N/A
cAGR	78.06	66.62	55.65	**79.63**	70.74	76.62
cCON	73.17	65.09	60.25	**74.21**	61.73	71.46
cOPN	**82.18**	76.08	74.18	81.56	79.47	81.29

Bold numbers represent the highest accuracy in given personality traits

Now, goal is to estimate likely performance of a model on out-of-sample data, so we applied k-fold cross-validation on the whole data to reduce the chances of overfitting of the data or high variance. We can get a more correct results of out-of-sample accuracy and effective data with k-fold cross-validation (every observation is given for both training and testing).With the cross-validation, we get results as minimum 55.03% and maximum cross-validation score as 84.29% for the Extraversion trait with ten folds. The average cross-validation score is 77.08% for EXT, 81.47% for OPN, 78.80% for AGR, and 74.22% for CON that we achieved with the k-fold cross-validation on the complete dataset. SVM and logistic regression are the most accurate and exact algorithms for our proposed research. Using all criteria, it produced great results for all attributes. SVM and logistic regression obtained maximum accuracy (82.18%) for cOPN trait. It has the best performance in all four aspects and all metrics. There are only four MBTI traits and five traits in the big five personality traits dataset. So, we could not achieve better accuracy NEU trait. We achieved 58.91% accuracy for NEU traits based on the big five personality traits dataset but better accuracy for the remaining four traits with the help of the MBTI dataset.

The comparison of present work with the previous research work Rahman et al. [2] and Majumder et al. [6] of personality traits classification (Extraversion, Conscientiousness, Agreeableness, Neuroticism, Openness) is given below: (Fig. 3).

6 Conclusion

Using several machine learning models, this study provided a method for determining the optimum personality characteristic identification methodology. We are creating model which can help recruiters to analyze the emotions of candidates as candidate screening method which will eventually save a lot of time of recruiter and will get alternative to traditional interview process. This research will surely help interviewers and companies in the online interview process where company could not analyze the emotions and personality traits of interviewee. This method consists of data filtering, data preprocessing, data merging, re-sampling of the data, and classification modeling. First, we merged the big 5 dataset and the MBTI dataset.

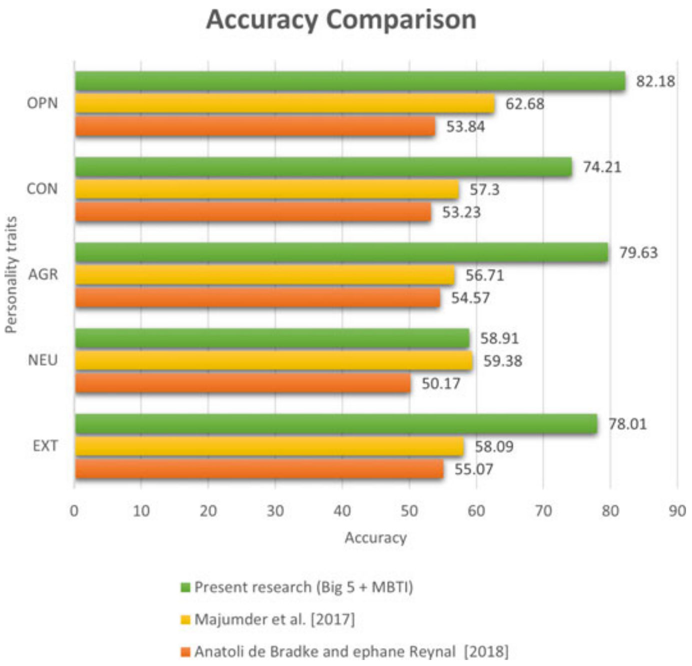


Fig. 3 Comparison of accuracy with present work

Following that, CountVectorizer and TF-IDF vectorizer were used to vectorize the processed data and then trained with support vector machine, logistic regression, random forest, decision trees, and Naive Bayes models. For comparative experimental analysis, five personality traits were used: EXT, NEU, AGR, CON, and OPN. Support vector machine and logistic regression outperformed other machine learning models, according to the comparative results. We observed logistic regression and support vector machine model give higher accuracy for the binary classification of the text data into personality traits with an accuracy score 78.01% for EXT, 79.63% for AGR, 58.91% for NEU, 74.21% for CON, and 81.56% for OPN traits. However, the Naive Bayes algorithm resulted in overall lower performance.

For future work, we plan to investigate the impact of deep learning techniques with neural networks for better accuracy. The personality traits recognition can be extended to audio and video emotion recognition with the help of convolutional neural networks and speech recognition methods.

References

1. Bajic E (2015) How the mbti can help you build a stronger company—forbes, 2015. URL https://www.forbes.com/sites/elenabajic/2015/09/28/how-the-mbti-can-help-you-build-a-stronger-company/?sh=51754881d93c. [Online]. Accessed 1 Sept 2021
2. Rahman MA, Al FaisalA, Khanam T, Amjad M, Siddik MS (2019) Personality detection from text using convolutional neural network. In: 2019 1st international conference on advances in science, engineering and robotics technology (ICASERT), pp 1–6. IEEE
3. Pramodh KC, Vijayalata Y (2016) Automatic personality recognition of authors using big five factor model. In: 2016 IEEE international conference on advances in computer applications (ICACA), pp 32–37. IEEE
4. Tareaf RB, Alhosseini SA, Berger P, Hennig P, Meinel C (2019) Towards automatic personality prediction using facebook likes metadata. In: 2019 IEEE 14th international conference on intelligent systems and knowledge engineering (ISKE), pp 714–719. IEEE
5. Lederman R, de Bradke A, Fabien M, Reynal S (2018) Multimodal emotion recognition. Projet Fil Rouge 2018–2019. URL https://www.overleaf.com/project/5c06f9e12aee1927b458fc4a
6. Majumder N, Poria S, Gelbukh A, Cambria E (2017) Deep learning-based document modeling for personality detection from text. IEEE Intell Syst 32(2):74–79
7. Abyaa A, Idrissi MK, Bennani S (2018) Predicting the learner's personality from educational data using supervised learning. In: Proceedings of the 12th international conference on intelligent systems: theories and applications, pp 1–7
8. Tinwala W, Rauniyar S (2021) Big five personality detection using deep convolutional neural networks
9. Kumar KNP, Gavrilova ML (2019) Personality traits classification on twitter. In: 2019 16th IEEE international conference on advanced video and signal based surveillance (AVSS), pp 1–8. IEEE
10. Kaggle. Myers-briggs personality type dataset (2017) URL https://www.kaggle.com/datasnaek/mbti-type. [Online]. Accessed 16 Aug 2021
11. Myers IB (1962) The myers-briggs type indicator: manual (1962)
12. Costa Jr PT, McCrae RR (2008) The revised neo personality inventory (neo-pi-r). Sage Publications, Inc
13. Kosinski M, Matz SC, Gosling SD, Popov V, Stillwell D (2015) Facebook as a research tool for the social sciences: opportunities, challenges, ethical considerations, and practical guidelines. American psychol 70(6):543
14. Pennebaker JW, King LA (1999) Linguistic styles: language use as an individual difference. J Pers Soc Psychol 77(6):296
15. Mairesse F, Walker MA, Mehl MR, Moore RK (2007) Using linguistic cues for the automatic recognition of personality in conversation and text. J Artif Intel Res 30:457–500
16. Furnham A (1996) The big five versus the big four: the relationship between the myers-briggs type indicator (mbti) and neo-pi five factor model of personality. Pers Individ Differ 21(2):303–307, ISSN 0191–8869. https://doi.org/10.1016/0191-8869(96)00033-5. URL https://www.sciencedirect.com/science/article/pii/0191886996000335
17. GeeksforGeeks. Svm hyperparameter tuning using gridsearchcv ml—geeksforgeeks 2019. URL https://en.wikipedia.org/wiki/Support-vector machine# Bayesian SVM. [Online]. Accessed 2 Sept 2021
18. KDnuggets. Algorithms, decision trees, explained. URL https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html. [Online]. Accessed 3 Sept 2021
19. Raj A (2021) Perfect recipe for classification using logistic regression, 2020. URL https://towardsdatascience.com/the-perfect-recipe-for-classification-using-logistic-regression-f8648e267592. [Online]. Accessed 3 Sept 2021]

Face Identification Through Facial Skeletal Features

R. Patel Bhautika and A. Desai Apurva

Abstract Automatic face identification technology has received lots of attention in the last few decades in the field of image analysis and computer vision. Automatic face identification is a challenging problem in image analysis. Psychologists and neurologists analyze facial images to discover human behavior and personality. Computer scientists deal with computation like feature extraction, similarity measurement, classification, etc., on facial images. Statistical shape analysis is one of the crucial areas in computer vision. Various face features are collected and then used for identification purposes or to find the asymmetry between the two parts of the face. The proposed study presents an algorithm for the face identification model using the distance between various skeletal facial features like mouth, eyes, and nose. For finding the facial features, the Viola–Jones face detection algorithm is used. K-nearest neighbor (KNN), Naive Bayes, and Bootstrap aggregation classifiers are used for the purpose of identification. The proposed model has been tested on the FEI, Faces94, Faces95, Grimace, BioID, and CVL datasets. The proposed algorithm is implemented in MATLAB and gives 94.33% accuracy.

Keywords Computer vision · Viola-Jones · Face identification · Nearest neighbor · Naive Bayes · BAG

1 Introduction

Authenticating the identity of a human being is a critical element in any security system. Biometrics is the use of unique physical or behavioral human characteristics to correctly name a person and the most reliable technique to answer the vast need of our society today. The term biometric comes from the Greek words bios (life) and

R. P. Bhautika
Smt. Tanuben and Dr. Manubhai Trivedi College of Information Science, Surat, India

A. D. Apurva (✉)
Department of Computer Science, Veer Narmad South Gujarat University, Surat, India
e-mail: aadesai@vnsgu.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_66

metrics (measure), as given by Sinha and Patil [1]. We, as human beings, intuitively use somebody's characteristics to recognize each other in our day-to-day life. Even a newborn child remembers her mother from the body characteristics of his mother.

Identification can be made using three ways, something you have (like, card, token, etc.), something you remember (like PIN, password, etc.) or something you are. Authentication using something you are is called biometric-based identification. A biometric system is a pattern recognition system that identifies a person by their different physical or behavioral characteristics. It refers to the automatic recognition of individuals based on their physiological and behavioral characteristics. By using biometrics, it is possible to prove a person's identity based on who you are rather than by what you have or what you remember, according to [2]. Human uses body features such as a face, voice, and gait for thousands of years to recognize each other. The technology has made use of such biometric methods for secure identification and verification. Jain et al. [3] states that human's physical as well as behavioral traits can be used for identification if it satisfies the characteristics like universality, distinctiveness, permanence, collectability, performance, acceptability, and circumvention.

The proposed algorithm uses the most significant facial skeletal features like eyes, lips, and nose. These are the key landmarks for face identification since these features are constant among the people and hence are suitable for identification even after a person has gone through facial surgery, and hence, recognition becomes more reliable. The proposed model work with faces having low illumination, rotation, variation in facial expressions, faces with a beard, makeover, and a person wearing glass and gives better results.

2 Literature Review

Face identification has become an active research area in recent times. Many researchers have contributed their ideas in biometric face recognition using different techniques. Face detection and recognition in an image become a significant work in the field of computer vision.

Teja and Ravi [4] proposed a face recognition system based on the subspaces technique. They first pre-processed the image to correct the illumination condition and then employed the feature extraction technique. The effective features are extracted by using the bi-directional principal component analysis (PCA) and linear discriminant analysis (LDA). They suggested Fisher's linear discriminant as a classifier on the FERET dataset and reported 87.16% recognition accuracy.

Face recognition system using discrete wavelet transforms (DWT) by decomposition of local features has been proposed by Patil and Boregowda [5]. They claim that the recognition rate has improved because of the use of the de-correlation of local features using DWT. In this suggested algorithm, the Gabor filters have been used for face recognition. The authors have tested the algorithm's performance on the FERET dataset and obtained approximately 90% of the accuracy. This algorithm outperforms the performance of the algorithm proposed by Teja and Ravi [4].

A face recognition algorithm that uses features derived from discrete cosine transform (DCT) coefficients and self-organizing map (SOM)-based classifier has been developed by Nagi et al. [6]. They have used DCT to extract features. The system is implemented in MATLAB and tested on an image database of 25 face images having five subjects and each subject having five images with different facial expressions. A vital variation here is that the testing images are the faces of different expressions. The suggested algorithm reports an 81.36% recognition rate.

Face detection from face images with diverse backgrounds is proposed by Jindal and Kumar [7]. The back-propagation feed-forward artificial neural networks (ANN) is used as a classifier, and the feature extraction is done using PCA. The ANN network is trained using 49 images of nine persons under different lighting conditions, facial expressions, hairstyles, and viewing conditions. Excellent accuracy of 95.45% is reported on the dataset of face images taken under varied conditions.

Image-based face detection has been developed by Ahmad et al. [8]; they have used AdaBoost and support vector machine (SVM) classifiers and had received around 90% accuracy on the five different datasets, namely Faces94, Faces95, Faces96, Grimace, and pain expression. They have also performed face recognition on the same database, and the average recognition rate using PCA was 71.15%, LDA was 77.90%, LBP was 82.94%, and using Gabor reached 92.35%.

Face part extraction using the face recognition system's segmentation method has been proposed by Dewi et al. [9], and they have extracted the eight unique facial features. Based on the experiment's result on 150 face images, the percentage of three face part distances reached 93.33%, while for eight faces, component distances increased to 100%. These eight face part distances form the face characteristic model through eigenvalue and eigenvector. However, the recognition using these face components has not been done by them.

Kanan et al. [10] proposed a morphometric analysis of Gujarati males. They find the facial index of a face using a formula (face length/face height × 100) and use this measure to categorize the adult male into one of the categories Hypereuriproscopic, Euriprosopic, Mesoprosopic, Leptoprosopic, or Hyperleptoprosopic. The study shows that the average facial index of Gujarati male is 81.7% at the same time; it presents that most of the Gujarati male faces are Euriprosopic, while Leptoprosopic males are seldom found.

The most crucial step in face identification using facial feature is the detection of various facial features like nose, eyes, and mouth. Landmark localization and measurement of the facial parts has been developed by Maghraby et al. [11] using the Viola–Jones and geometric approaches. To improve a facial feature's detection rate, they have divided the whole face into different regions and then find the related feature only in that region instead of finding it in the entire face. The recognition accuracy for detecting various facial features like eye-pair, nose, and mouth is 98.98%. Apart from this, the identification of the person using these measurements stays untouched.

Face recognition using neural networks has been proposed by Nandini et al. [12]. In this system, the recognition was done by comparing the new face's characteristics to that of known individuals. In feature extraction, the distance between eyeballs and mouth end-points is calculated. The recognition is performed by neural network (NN) using back-propagation networks (BPN) and radial basis function (RBF) networks. The algorithm is tested on a total of 90 images, 64 for training, and the rest of the 26 images for testing. The recognition rate using BPN was 96.66%, and that using BPN + RBF was 98.88%.

To improve face recognition accuracy, a hybrid approach that combines artificial neural network (ANN), and linear Fisher discriminant analysis (LDA) was proposed by Parveen et al. [13]. They have proposed weighted feature selection to improve face recognition accuracy and got 91.43% accuracy.

Face recognition using a bidirectional neural network was proposed by Mazloum et al. [14]. In addition to each person's information, the variance in facial expressions is also considered for recognition. To improve recognition, they have applied an inverse network. The algorithm was tested on the AUT database with an accuracy rate of 92.5%.

A review on different classifiers used in face recognition under pose and illumination variation has been conducted by Rajalakshmiand and Jeyakumar [15]. They have claimed that the recognition rate using PCA was 90.5%, and using ICA, it was 92.5% for pose variation with large rotation. The acceptance rate for PCA was 90.5%, and using ICA, it was 86.5% for variation in illumination condition.

Assadi and Behrad [16] have proposed human face recognition using texture and depth information to overcome pose variation and illumination. The algorithm was tested on the FRAV3D database with 106 images and reported accuracy rate of 88.96%.

3 Proposed Model

In this section, a detailed description of the proposed model is presented. There are two phases in the biometric system: a learning phase and a recognition phase, according to Dorizzi [17]. The proposed model is divided into three modules, (i) Feature extraction, (ii) Training, and (iii) Identification module.

3.1 Feature Extraction Module

The following steps are applied to find facial skeletal features graphically represented in Fig. 1.

Step 1: Find the frontal face from a given image using the CascaseObjectDetection system object that uses the Viola–Jones object detection framework. This framework was introduced by Paul Viola and Michael Jones.

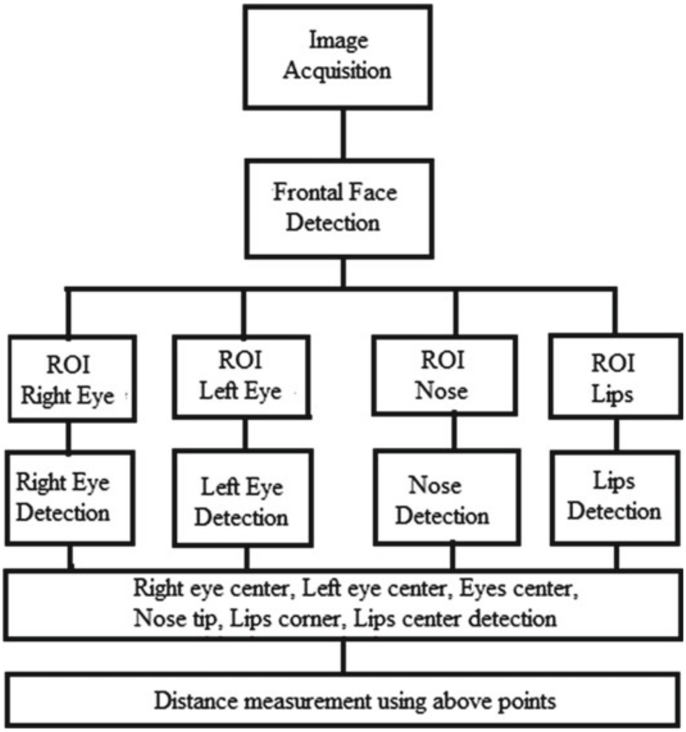
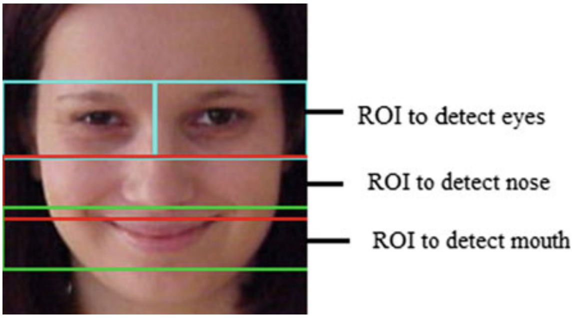


Fig. 1 Graphical view of the feature extraction module

Step 2: From the human frontal face structure, eyes, nose, and mouth areas are situated in the upper, middle, and lower portion of the face. Figure 2 shows the region of interest (ROI) for various facial landmarks.

Step 3: Find the right and left eye, respectively, and then find out the left and right eye pupils. The proposed system uses the iris recognition system, which consists of an automatic segmentation system based on Hough transform to localize the circular

Fig. 2 Region of interest to detect face parts



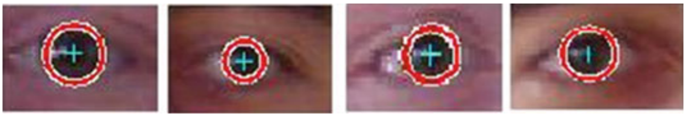


Fig. 3 Right eye and left eye center detection

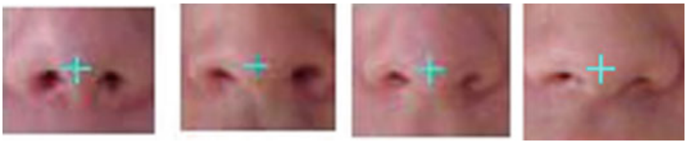


Fig. 4 Nose tip detection samples

iris pupil detection as proposed by Maghraby et al. [11]. Figure 3 shows the result of eye center detection.

Step 4: Locate the nose using the Viola–Jones nose detector, which gives us a rectangular bounding box. After that find out nose tip using the Formula 1. Where (x1, y1) and (x2, y2) are top left and bottom right corners of rectangle. Figure 4 shows the result of nose tip detection.

$$\left(\frac{(x1 + y1)}{2}, \frac{(x2 + y2)}{2}\right) \tag{1}$$

Step 5: Find mouth bounding box and then the corners of the mouth by applying the following steps. The result of this process is shown in Fig. 5.

(i) Convert RGB image into YCbCr color space.

(ii) Perform the fusion of RGB image and YCbCr image so that the lip region will get more energy, and edges will be easily detected presented by Saeed and Dugelay [18].

(iii) Find out the global threshold using Otsu's method and apply it.

(iv) Find out the edges of the fused grayscale image using the Canny edge detector suggested by Nain et al. [19].

(v) Traverse the image from the top left corner to the bottom right and find the first-pixel with intensity 1, considered one corner. Similarly, traverse the image from top right to bottom left and find the first-pixel with intensity 1, which is considered another corner.

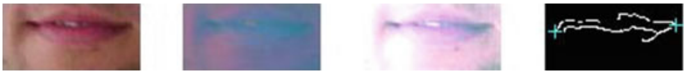


Fig. 5 Lip corners detection: original RGB image, YCbCr representation, fusion of RGB and YCbCr, and lips with edge and corners detection, respectively

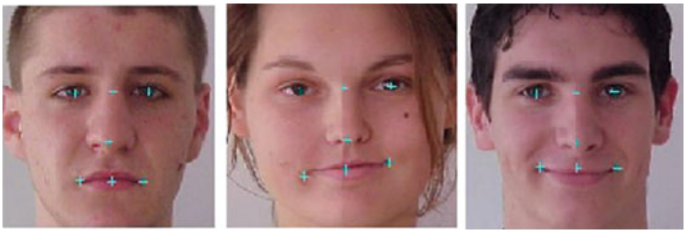


Fig. 6 Sample images annotated with different points used for feature extraction

Step 6: Find the distinctive features between the seven points as shown in Fig. 6. P1 = Right eye center, P2 = Left eye center, P3 = Center between two eyes, P4 = Nose tip, P5 = Right corner of the mouth, P6 = Left corner of the mouth, P7 = Center of mouth.

Step 7: The representative features are extracted by computing distance between these points, which are used for identification. Some sample images after feature extractions where the features are extracted from the images having varied conditions, like good illumination, low illumination, a person wearing glass, and a person with makeover are present in Fig. 7.

The following features are used for identification in the proposed algorithm.

F1 = Distance between points P2 and P3.
F2 = Distance between points P1 and P3.
F3 = Distance between points P2 and P7.
F4 = Distance between points P1 and P7.
F5 = Distance between points P2 and P4.
F6 = Distance between points P1 and P4.
F7 = Distance between left eye center and center of F3.
F8 = Distance between right eye center and center of F4.
F9 = Distance between left eye center and center of F6.
F10 = Distance between right eye center and center of F5.
F11 = Distance between points P3 and P4.
F12 = Distance between points P3 and P7.
F13 = Radius of left eye.
F14 = Radius of right eye.

The Euclidian distance has been utilized for finding the distance between two points, as shown in the Eq. (2).

$$d(x, y) = \left(\sqrt{\sum_{i=1}^{n}(Xi - Yi)^2}\right) \tag{2}$$

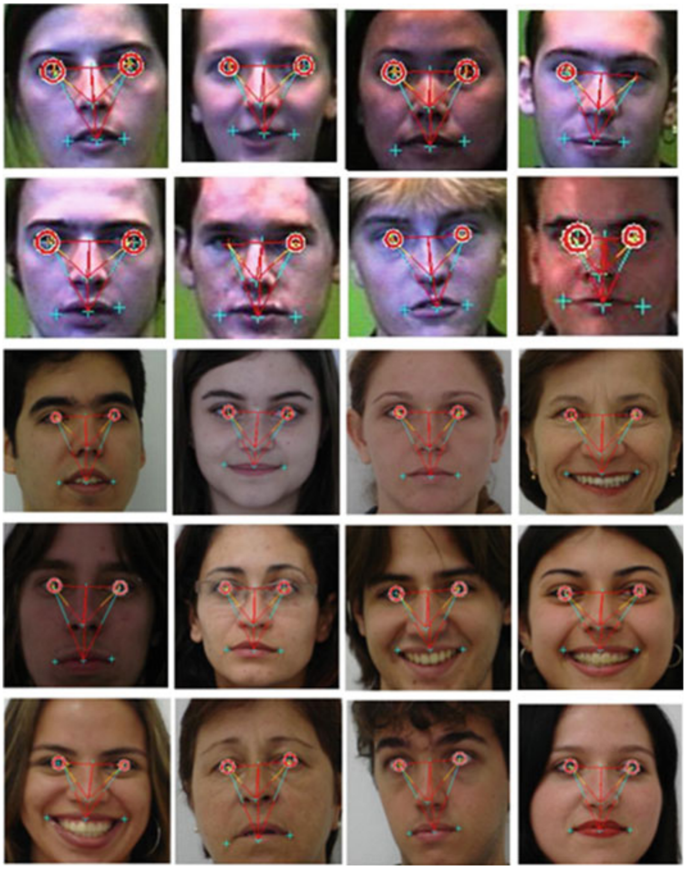


Fig. 7 Feature extraction sample

3.2 Training Module

The proposed system is trained and tested on [20–25] datasets. Table 1 shows the vector size of the training and testing of each dataset.

FEI dataset contains images taken against a white homogenous background in an upright frontal position with profile rotation of up to about 180°. Faces94 dataset includes images taken under plain background, minor variation in head turn and slant, considerable expression changes, and no lighting condition variation. Faces95 datasets contain images with uniform background, large head scale variation, a notable change in lighting, and some variation in expression. Images in the Grimace dataset are taken under plain background, small head rotation, and major variation in expression. BioID dataset includes images having a large variation in illumination, background, and face size. CVL dataset is prepared with a projection screen in the background, and all images are having uniform illumination and no flash.

Table 1 Training and testing vector size

Dataset name	No. of individual	Training vector size	Testing vector size
FEI	200	600	400
Faces94	153	2295	459
Faces95	72	1080	216
Grimace	18	270	54
BioID	23	897	69
CVL	114	228	114

3.3 Identification Module

We have fused KNN, Naive Bayes, and bootstrap aggregation classifiers to improve face recognition accuracy for identification. Figure 8 shows the face recognition process used by the proposed system. Further, in the KNN, we have used 'Euclidean' distance to find the nearest. The Naive Bayes classifier classifies an object by calculating the conditional probabilities for all classes. The bootstrap aggregation classifier uses the ensemble method that combines the predictions from multiple machine learning algorithms to make more accurate predictions. Figure 9 shows the sample images used for training and identification.

4 Experimental Result

The work presented in this paper uses six different datasets available publicly to test the accuracy of the proposed system. Table 2 represents the accuracy of our proposed approach on each dataset.

The proposed system gives efficient results compared to other face recognition approaches. To see the effectiveness, we have compared our face recognition approach with different face recognition algorithms we have referred in the literature review, and the result is shown in Fig. 10.

5 Conclusion and Future Work

In this paper, we have presented how skeletal facial features are extracted and used for the identification of a person using the Viola–Jones CascadeObjectDetector system object along with different classifiers. Automatic face identification based on the skeletal features of a face is designed to recognize the person. We have used skeletal features as they are not changeable. The proposed system is evaluated on diverse types of images containing faces of different genders and sizes. All these images are

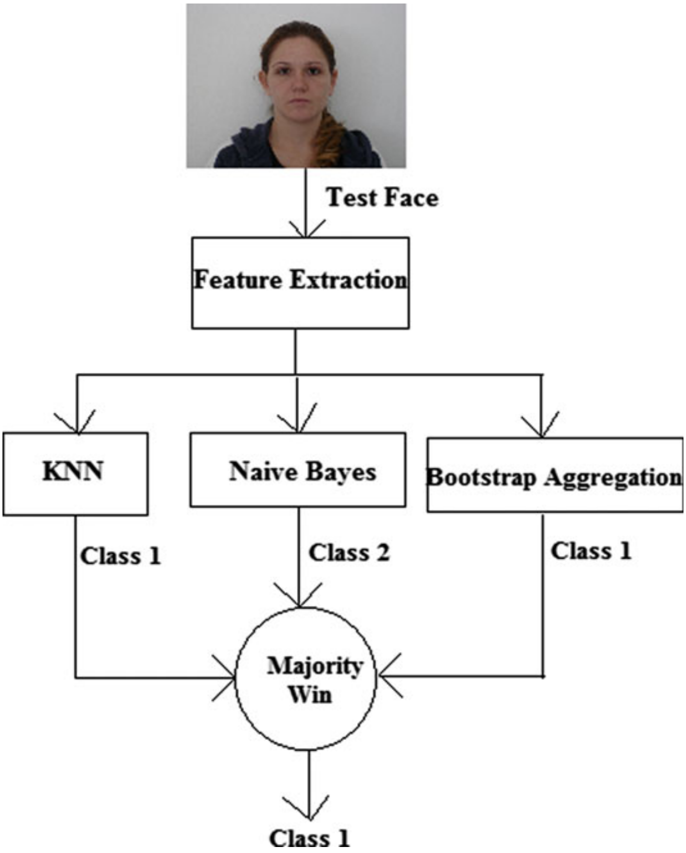


Fig. 8 Face identification process

taken under different environmental conditions. The experimental result shows that the performance of the face recognition system increases as the size of the training dataset increases. The accuracy of the algorithm with the FEI and CVL database is comparatively low than the rest of the databases. This is because only 3 and 2 images of everyone are used for training the system. At the same time, the result is high in other databases because around 15 images are used for training the system. In future, we will try to perform face recognition from real-time video.

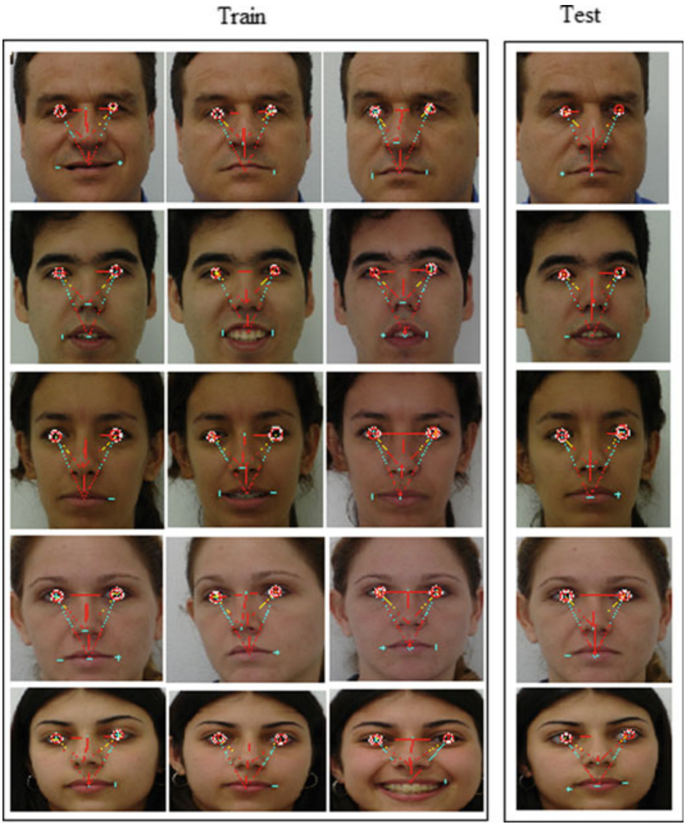


Fig. 9 Result of training and identification

Table 2 Face identification accuracy

S. No	Dataset	Accuracy (%)
1	FEI	82.33
2	Faces94	94.12
3	Faces95	90.28
4	Grimace	94.33
5	BioID	91.30
6	CVL	70.02

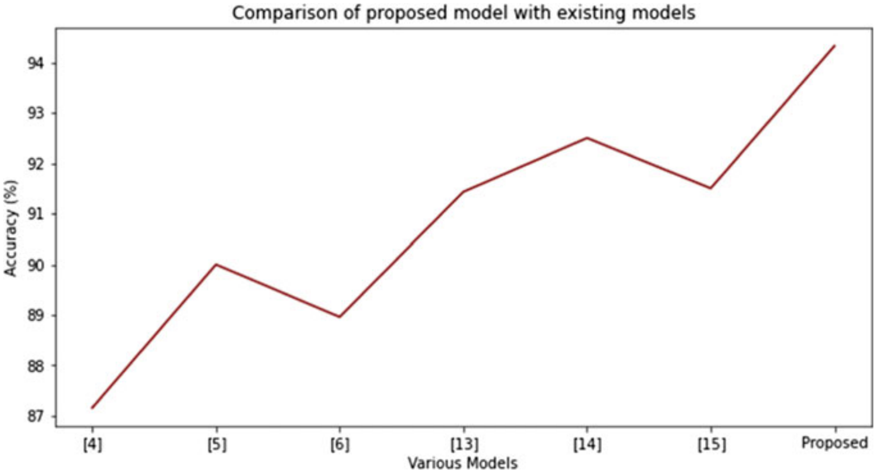


Fig. 10 Comparison of proposed system with other system

Acknowledgements The authors are thankful to UGC for the SAP-II (No. F.3-48/2011(SAP-II), under which this research work is carried out.

References

1. Sinha GR, Patil SB (2013) Biometrics-concepts and applications. Wiley
2. Delac K, Grgic M (2004) A survey of biometric recognition methods. In: 46th international symposium electronics in marine, pp 184–193
3. Jain AK, Ross A, Prabhakar S (2004) An introduction to biometric recognition. IEEE Trans Circuits Syst Video Technol Special Issue Image Video-Based Biometrics 14(1):1–29. https://doi.org/10.1109/TCSVT.2003.818349
4. Teja GP, Ravi S (2012) Face recognition using subspaces techniques. In: International conference on recent trends in information technology, pp 103-107. https://doi.org/10.1109/ICRTIT.2012.6206780
5. Patil NK, Vasudha S, Boregowda LR (2013) Performance improvement of face recognition system by decomposition of local features using discrete wavelet transforms. In: International symposium on electronic system design, pp 172-176. https://doi.org/10.1109/ISED.2013.41
6. Nagi J, Ahmed SK, Nagi F (2008) A MATLAB based face recognition system using image processing and neural networks. In: 4th international colloquium on signal processing and its applications, Kuala Lumpur, Malaysia, pp 83–88
7. Jindal N, Kumar V (2013) Enhanced face recognition algorithm using PCA with artificial neural networks. Int J Adv Res Comput Sci Softw Eng 3(6):864–872
8. Ahmad F, Najam A, Ahmed Z (2012) Image-based face detection and recognition: state of the art. Int J Comput Sci Issues 9(6):169–172
9. Dewi AR, Adang S, Madenda S, Suryadi HS (2011) Face component extraction using segmentation method on face recognition system. J Emerg Trends Comput Inf Sci 2(2):67–72
10. Kanan U, Gandotra A, Desai A, Andani R (2012) Variation in facial index of Gujarati males-a photometric study. Int J Med Health Sci 1(4):27–31

11. Maghraby AE, Abdalla M, Enany O, El Y (2014) Detect and analyze face parts information using Viola-Jones and geometric approaches. Int J Comput Appl 101(3):23–28. https://doi.org/10.5120/17667-8494
12. Nandini M, Bhargavi P, Raja Sekhar G (2013) Face recognition using neural network. Int J Sci Res Publ 3(3):1–5
13. Parveen P, Thuraisingham B (2006) Face recognition using various classifiers: ANN, linear discriminant and PCA. Technical Report. Department of Computer Science, University of Texas, pp 2–15
14. Mazloum J, Jalali A, Amiryan J (2012) A novel bidirectional neural network for face recognition. In: 2nd international econference on computer and knowledge engineering (ICCKE), pp 18–23. https://doi.org/10.1109/ICCKE.2012.6395345
15. Rajalakshmiand R, Jeyakumar MK (2012) A review on classifiers used in face recognition methods under pose and illumination variation. Int J Comput Sci Issues 9(6):474–485
16. Assadi A, Behrad A (2010) A new method for human face recognition using texture and depth information. In: 10th symposium on neural network applications in electrical engineering, pp 201–205. https://doi.org/10.1109/NEUREL.2010.5644065
17. Dorizzi B (2005) Biometrics at the frontiers, assessing the impact on society technical impact of biometrics. Background paper for the institute of prospective technological studies, DG JRC—Sevilla, European Commission, pp 1–22
18. Saeed U, Dugelay J (2010) Combining edge detection and region segmentation for lip contour extraction. In: International conference on articulated motion and deformable objects, pp 11-20. https://doi.org/10.1007/978-3-642-14061-7_2
19. Nain N, Jindal G, Garg A, Jain A (2008) Dynamic thresholding based edge detection. In: Proceeding of the world congress on Engineering 1
20. FEI face database. (n.d.). Retrieved from https://data.fei.org
21. Faces94 face database. (n.d.). Retrieved from http://cswww.essex.ac.uk/mv/allfaces/faces94.html
22. Faces95 face database. (n.d.). Retrieved from http://cswww.essex.ac.uk/mv/allfaces/faces95.html
23. Grimace face database. (n.d.). Retrieved from http://cswww.essex.ac.uk/mv/allfaces/grimace.html
24. BioIDface database. (n.d.). Retrieved from https://ftp.uni-erlangen.de/pub/facedb
25. Peer P CVL face database. (n.d.). Retrieved from http://www.lrv.fri.uni-lj.si/facedb.html

Machine Learning-Based Model for Effective Resource Provisioning in Cloud

Payal Saluja, Swati Jain, and Madhuri Bhavsar

Abstract Ever increasing demand of cloud infrastructure services and Global adoption of Cloud Computing by diverse organizations has led to establishment of data centers with huge compute and storage infrastructures. Cloud suffices the consumer requirements by providing on demand access to the resources. Software components of cloud architecture work synchronously in order to fulfill the cloud consumer demands. Cloud resource management is one of the key functionalities that enables seamless access to cloud resources. The resource management in cloud is done considering several QoS parameters like workload balancing, resource utilization, application performance, energy efficiency. Several algorithms and techniques have been developed to target one or two of the above parameters however due to dynamic nature of cloud it is the need of the hour to develop smarter mechanisms that can lead to efficient management of resources in cloud. In this paper, we analyze and compare the clustering-based machine learning (ML) approaches that will help cloud resource manager for efficient management of cloud resources. The paper also proposes the system architecture for integrating ML-based algorithms for cloud resource allocation.

Keywords Cloud resource provisioning · Machine learning · Clustering · Workload management

Supported by Nirma University. https://cse.nirmauni.ac.in/.

P. Saluja · S. Jain · M. Bhavsar (✉)
Department of Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad, Gujarat, India
e-mail: madhuri.bhavsar@nirmauni.ac.in

P. Saluja
e-mail: payalsaluja2@gmail.com

S. Jain
e-mail: swati.jain@nirmauni.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_67

1 Introduction

Cloud computing is a form of utility computing that provisions computational, storage and network resources as a metered service [1]. The easy access to its services is enabled by its characteristics like self-provisioning, web enabled access, multi-tenancy, and elasticity. This model of computing is a perfect fit for the modern compute intensive and data-intensive research. Cloud consists of many software components that work behind the scenes to provide seamless access to the cloud resources. Cloud resource provisioning is one of the key functionalities of cloud computing that enable seamless access to cloud users to scientific users [2]. Providing the optimized amount of cloud resources leads to effective utilization of cloud resources as well as fulfills the timely needs of cloud consumers without impacting the hosted applications on cloud. The three prime functionalities required by a good resource management in cloud includes: resource provisioning, resource scheduling, and resource monitoring [3]. Resource provisioning involves searching and allocating the suitable resources for executing a workload. Resource scheduling is generating the sequence or schedule for assigning resources to tasks. Resource monitoring is observing or reviewing the health and utilization of cloud resources.

Resource provisioning is a complex task in highly dynamic environments like cloud. Considering the various performance parameters of cloud like load balancing, resource utilization, application performance its necessary to have an efficient and smart cloud resource manager.

This paper details about the challenges of resource provisioning in the highly dynamic environment like cloud and discusses the clustering-based machine learning approaches that would enable cloud resource manager to provide an efficient resource allocation. Authors also proposes a system architecture that integrates machine learning approaches in cloud stack for better resource provisioning. It aims to build a machine learning-based model to classify the workload of user requests. This classification of the workload would enable the cloud resource broker to allocate the sufficient resource for handling the workload request.

2 Background

2.1 Complexity of Resource Allocation in Cloud

The resource allocation in cloud is a complex optimization problem that needs consideration of multiple parameters in the dynamic environment of cloud [4]. However, reaching an optimal solution for each and every consumer request along with effective utilization of cloud resources is close to impossible. The traditional methodologies and algorithms often fail to handle the dynamism and the scale of the cloud leading to lower Quality of services (QoS).

2.2 Role of Machine Learning Approaches

Machine learning is a subfield of artificial intelligence that results in gaining knowledge from the historical data [5]. It builds the model based on previous data that basically trains the model that predicts the outcome when new requests comes It is a multidisciplinary field that needs knowledge of application specific domain, probability, statistics, computer science, artificial intelligence. It enables effective analysis and predictions based on the models that have been built using the historical data. This approach would be very useful for resource provisioning in cloud as the model trained based on the previous workloads can be used to suggest the resource allocation for the fresh incoming workloads.

2.3 Related Work

Resource allocation is an important aspect of provisioning cloud computing services. Several approaches have been worked upon to achieve effective resource allocation like scheduling algorithms, load balancing algorithms, clustering algorithms, etc. Also, research works have been conducted to conduct comparative study of clustering and scheduling algorithms like K-means and First-in-First-out [6]. The study concluded with better results with K-means clustering in cloudsim environment. Another research have suggested the use of load balancing algorithms for virtual machine allocation across multiple data centers [7]. Also, research work has been conducted to survey VM allocation methodologies like FCFS, Round Robin for efficient resource allocation in cloud environments [8]. Authors also proposed resource allocation model considering factors like number of CPU cores, Memory capacity, and Hard Drive capacity of VMs to achieve the maximum resource utilization rate. Research work by M. Shah proposed use of load balancing algorithms for resource allocations and used Throttle Load Balancing Algorithm to estimate the response time of each virtual machine [9]. Another research has used K-means clustering algorithm for dynamic allocation of virtual machines and measured the performance in terms of cost in data center. Here, based on the clustering, workload is allocated to a data center [10]. Some of the research works have targeted efficient load balancing considering computational capabilities of resources and energy efficiency. Research titled "RALBA" proposed a novel load balancing Algorithm to ensure a balanced distribution of workload based on computation capabilities of resources [11]. The authors evaluated RALBA and other eight scheduling heuristics in cloud computing in terms of resource utilization. Another research work describes the virtual machine allocation using K-means clustering algorithms considering energy savings as the QoS parameter [12]. With the background of results of various clustering-based research works conducted, the authors of the paper have compared the performance based on cluster quality metrics of various clustering-based machine learning approaches that could be utilized for classifying the cloud workload for further resource allocation.

3 Building ML-Based Model for Resource Allocation

3.1 Clustering-Based Machine Learning

Machine Learning algorithms are designed to build a model that learns by itself by execution of a particular task, again and again on different training data [13]. The built model is capable of future predictions based on the optimal parameter selection and optimization. It is applicable to a wide range of domains and there are 4 major approaches that are used: Supervised learning (with labeled training data), unsupervised learning (no labeled data), Semi-supervised learning and Reinforcement learning. The approach that we use in order to train the model depends upon the required outcome as well as the kind of training dataset that is available with us. The dataset availability for real time cloud workload is usually in terms of job size (MIPS), CPU utilization, memory and storage usage, etc which is under the category of unlabelled data. Such unlabelled data is used to train model with unsupervised ML approach. Clustering is the most adopted unsupervised machine learning technique for data mining and analysis. In this, data points are grouped together based on their euclidean or probabilistic distance to find the similarity of the data points. Clustering enables data classification and dimensionality reduction of the feature space that helps to build cleaner and more accurate models.

3.2 Clustering for Resource Allocation

In scenario of cloud, there is a need for resource allocation of n independent heterogeneous tasks on m heterogeneous machines. Each task is of length L in terms of number of instructions (Million Instructions—MI) Each machine has a processing capacity in terms of million instructions per second (MIPS). The resource allocation is an optimization problem that is formulated such that n tasks need to be allocated to m machines in such a way that workload on the machines should be in a balanced state or all the machines should have equal workload according to their capacity. Also, the total completion time of the tasks should be reduced. Machine learning algorithms are useful in classifying the type of workload based on the job length. Hence, in this research we propose a system that makes use of clustering-based ML algorithms to form task clusters for better allocation of cloud resources. Clustering algorithms that we explored are discussed below in the next section.

3.3 Clustering Algorithms

Clustering is a data mining technique used for prediction of class of unlabeled objects. There are several clustering algorithms that are available like K-Means clustering, Mean Shift Clustering, DBSCAN Clustering, Gaussian Mixture models, BIRCH clustering [14]. Some of the clustering techniques are discussed as follows:

K-Means Clustering K-means clustering is a form of ‘unsupervised learning’ that is useful for grouping unlabeled data [15]. It works efficiently in handling datasets with a single attribute and is also fast and easy to implement. It has been used in data clustering in various domains like cluster-based task scheduling on computational resources load balancing in Cloud Data centers.

DBSCAN Clustering DBSCAN is a clustering algorithm that defines clusters as continuous regions of high density [16]. The advantage of DBSCAN over K-means is that it does not need number of clusters to be pre-defined, it is also capable of detecting the outliers. Also, it can cluster the data in any arbitrary shapes. In the case of DBSCAN, instead of guessing the number of clusters, we define two hyperparameters: epsilon and min Points to arrive at clusters. Epsilon(eps) demarcates the neighborhood around a data by calculating the distance, d between two points. If ($d <=$ eps), points are considered as neighbors and fall under single cluster MinPts: Minimum number of neighbors (data points) within eps radius.

BIRCH Clustering Balanced iterative Reducing and Clustering using Hierarchies (BIRCH) is a clustering algorithm that is capable for summarizing a larger data set into a smaller one at the same time preserving as much information as possible [17]. Many clustering algorithms face scalability challenge and this challenge is overcome by BIRCH algorithm. The advantages that it gives are the clustering is done in a incremental fashion covering all the data points. Also,the algorithm allows splitting of sub clusters making it faster.

4 Dataset Preparation

Algorithm developers of cloud resource scheduling consider several parameters like resource utilization, workload balancing, energy efficiency, response turnaround time. While developing these algorithms, the developer needs the cloud workloads or cloud usage datasets using which the models could be trained and validated. However, getting access to real cloud datasets is very difficult due to security and SLA clauses for user data that protects user confidentiality and privacy. Some of the available real workload traces like Google cluster traces [18], Yahoo cluster traces [19], Facebook Hadoop workload [20], OpenCloud Hadoop workload [21], Eucalyptus IaaS cloud Workload [22] etc. However, most of these are difficult to process and needs a number of preprocessing steps before using. Thus, we utilized the Google Cloud Job Dataset (GoCJ) that is already pre-processed and easy to use [23].

4.1 Dataset Description

Source: The reference of the dataset has been taken from MDPI, who is pioneer in scholarly open access publishing of research work research data across all disciplines. The data has been described and published in the research by the title "GoCJ: Google Cloud Jobs Dataset for Distributed and Cloud Computing Infrastructures". The dataset generation process has been done using the Monte Carlo (MC) simulation method and discussed in detail in the above. The dataset is available at Mendeley Data repository [24].

Why GoCJ: GoCJ dataset provides a similar workload trace as of Google cluster traces and Mapreduce logs [25]. The data looked very real and unbaised and we considered this to be very useful for training the ML model for workload prediction. It has also been suggested that GoCJ dataset can also be used developing validating algorithms and models for scheduling and resource allocation mechanisms in cloud computing.

Dataset Structure: The GoCJ dataset consists of 21 text files and 1 excel Monte Carlo (MC) data generator. Each dataset filename contains the count of the number of jobs for which million instructions has been listed.

Data Generation Process: A list of 50 different-size jobs are identified and input into MC bootstrapping as the original dataset. Each job size in the dataset is treated with equal probability in repeated sampling by bootstrapping. The sizes of jobs in the GoCJ dataset is presented in terms of MIs. The dataset is generated by bootstrapped MC simulation using an Excel worksheet is indicated in Table 1.

4.2 Final Dataset

A final dataset has been produced by summing up the MIPS of the individual workload file containing XXX jobs. An average of MIPS of each of the txt file is also calculated to produce the final dataset. Also, we generated set of text files containing upto 1500 jobs using excel generator file. Table 2 depicts the final dataset that has been prepared for training the model for clustering.

5 Results

As observed in the GoCJ dataset, the data is unlabeled so, unsupervised ML approaches need to be implemented for classification of the data set. In unsupervised machine learning, there are various methods for labeling such datasets. Clustering methods are most commonly used for labeling them. In this work, we have used various clustering algorithms to form the clusters of jobs with similar workload. We also aimed to do cluster analysis of the clusters formed for its goodness using the performance metrics for clustering achieved with K-means, DBSCAN Birch algorithms and suggest the algorithm that gives the best quality of clusters.

Table 1 Snapshot of the excel sheet dataset generator

Probability	Cumulative probability	Original job	Random no	GoCJ dataset
0.02	0	15,000	0.4493544	87,000
0.02	0.02	27,500	0.4567427	87,000
0.02	0.04	40,000	0.2049258	61,000
0.02	0.06	45,000	0.5095939	93,000
0.02	0.08	47,000	0.9023972	150,000
0.02	0.1	49,000	0.2492184	65,000
0.02	0.12	51,000	0.5934954	101,000
0.02	0.14	53,000	0.0069061	15,000
0.02	0.16	55,000	0.4380102	85,000
0.02	0.18	59,000	0.9914794	900,000
0.02	0.2	61,000	0.1037235	49,000
0.02	0.22	63,000	0.3543169	77,000
0.02	0.24	65,000	0.6183884	103,000
0.02	0.26	67,000	0.0770614	45,000
0.02	0.28	71,000	0.8634219	129,000
0.02	0.3	73,000	0.1471578	53,000
0.02	0.32	75,000	0.3229733	75,000
0.02	0.34	77,000	0.0222821	27,500
0.02	0.36	79,000	0.7381537	115,000
0.02	0.38	81,000	0.8165597	123,000
0.02	0.4	83,000	0.5247064	95,000
0.02	0.42	85,000	0.3259397	75,000
0.02	0.44	87,000	0.3930397	81,000

5.1 Results of Clustering Algorithm

Following are the results achieved for workload classification after implementing clustering algorithms considering two features—Total of Job sizes in a batch and Average job size:

k-means Clustering K-means clustering algorithm groups the similar data points together and forms the clusters represented by k. The value of k is optimally chosen using either Elbow method or Silhouette Method. Here, in this work we have made use of Elbow method for determining the optimal value of k. Elbow method picks up the range of values and takes the best value by calculating the sum of the square of the points and calculates the average distance. Figure 1 shows the data before applying the K-means clustering algorithm. Here all data points are unclassified and does not seem to be as per their classes. With this its very difficult to differentiate between various categories of workloads.

Table 2 Final dataset

No of jobs	Total MIPS	Avg MIPS	Min MIPS	Max MIPS
100	13,509,500	135,095	15,000	900,000
150	22,180,500	147,870	15,000	900,000
200	27,219,500	136,097.5	15,000	900,000
250	29,977,000	119,908	15,000	900,000
300	41,426,000	138,086.66	7000	900,000
350	47,268,000	135,051.42	15,000	900,000
400	51,869,500	129,673.75	15,000	900,000
450	26,632,000	59,182.22	15,000	900,000
500	65,001,500	130,003	15,000	900,000
550	71,658,000	130,287.27	15,000	900,000
600	82,251,000	137,085	15,000	900,000
650	88,506,000	136,163.07	15,000	900,000
700	87,493,500	124,990.71	15,000	900,000
750	102,055,000	136,073.33	15,000	900,000
800	98,958,000	123,697.5	15,000	900,000
850	106,353,500	125,121.76	15,000	900,000
900	120,047,000	133,385.55	15,000	900,000
1000	129,662,000	129,662	15,000	900,000
1050	133,401,500	127,049.04	15,000	900,000
1100	141,937,500	127,541	15,000	90,000
1200	156,367,000	128,870	15,000	90,000
1300	137,100,196	110,357	15,000	525,000
1400	143,728,691	102,663	15,000	525,000
1500	159,764,918	103,418	15,000	525,000

Fig. 1 Input dataset before applying K-means clustering

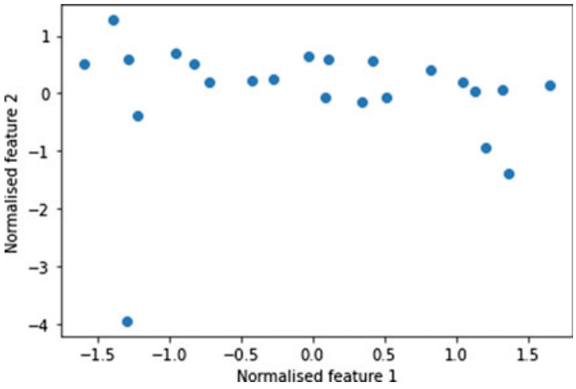


Fig. 2 Graph between K-values and the within-cluster sum of the square

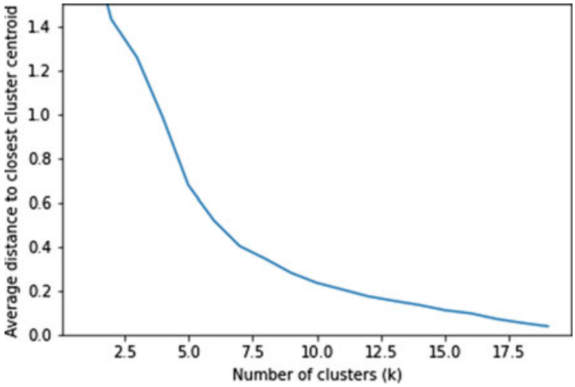


Fig. 3 Scatter plot of K-means clustering of dataset

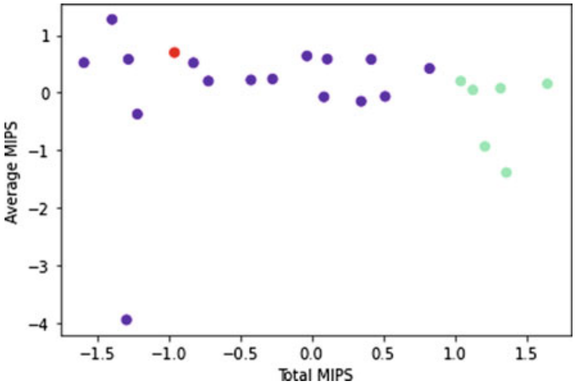


Figure 2 indicates a graph between K-values and the within-cluster sum of the square. At the value $k = 3$, the graph indicates an abrupt decrease in average distance and hence, this value of k is considered optimal value for K-means clustering algorithm.

The Scatter plot of final workload clusters is shown in Fig. 3. This represents the data clusters after applying the K-means clustering algorithm and we can see that the data has been clustered into three different cluster categories.

DBSCAN Clustering This section presents the clustering of the GoCJ dataset using the DBSCAN clustering algorithm. The algorithm makes use two important optimization hyper parameters, tuning which correctly provides the best performance of the algorithm. The scatter plot of the Input dataset points is shown in Fig. 4. The algorithm was executed for iterative values of epsilon with different combinations minimum points and following results were obtained as indicated in Table 3.

The algorithm was run for different values of epsilon and the resulting plots of the same are indicated in Fig. 5. DBSCAN results for different values of eps. It is observed that 4 clear clusters are formed, when DBSCAN clustering was done with epsilon value of 0.5.

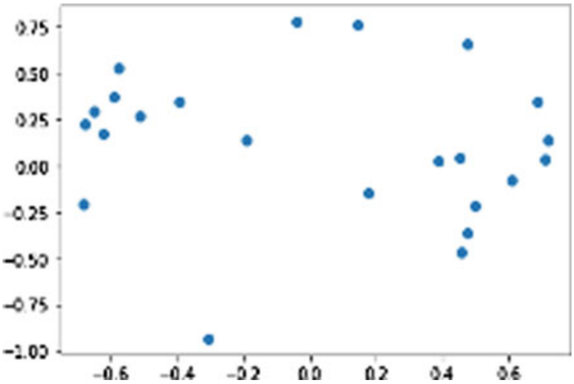


Fig. 4 Scatter Plot of input dataset before DBSCAN

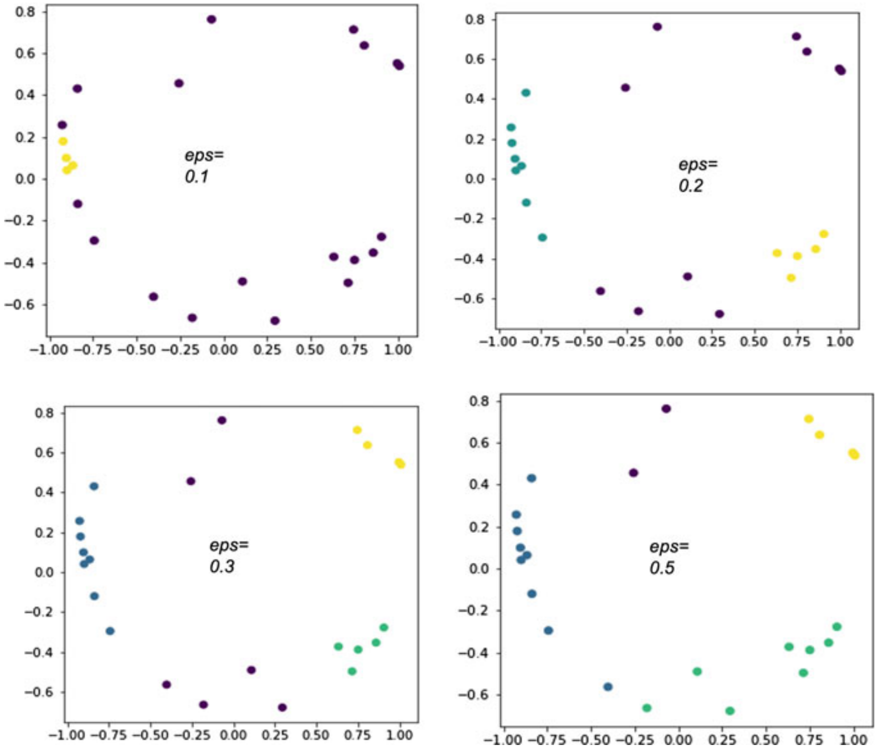


Fig. 5 Scatter plot of DBSCAN clustering of data set

Table 3 DBSCAN results for various combinations of epsilon minpoints

S. No.	Eps	Min points	SIL score	No. of clusters	Type of clusters
1	0.1	2	0.6059	5	Non clear clusters with very few points
2		3	0.4433	2	Non clear clusters with very few points
3		4	0.3881	2	Non clear clusters with very few points
4	0.2	2	0.7397	5	Non clear clusters with very few points
5		3	0.642	3	Non clear clusters with very few points
6		4	0.642	3	Non clear clusters with very few points
7	0.3	2	0.729	6	Non clear clusters with very few points
8		3	0.725	4	Non clear clusters with very few points
9		4	0.7250	4	Non clear clusters with very few points
10	0.4	2	0.796	5	Outliers
11		3	0.7914	5	Outliers
12		4	0.725	4	Outliers
13	0.5	2	0.4749	3	Clear clusters
14		3	0.4524	3	Clear clusters
15		4	0.7215	4	Clear clusters

5.2 Performance Analysis of Clusters

The clusters that get formed as an outcome of the clustering algorithms needs to be analyzed for their quality [26]. A cluster is a collection of 2 or more similar objects that are placed close to each other with some minimum distance. Clusters are evaluated based on some similarity or dissimilarity measure such as the distance between cluster points. A good clustering algorithm is one that produces high quality clusters with high similarity and low similarity between clusters. There are several metrics that are used to measure the quality of the clusters formed by clustering algorithms like Silhouette Score, Calinski Harabasz Score, Davies Bouldin index [27, 28]. Silhouette score is metric to measure goodness of a cluster whose value ranges from −1 (bad cluster) to 1 (good cluster). Davies Bouldin index is the similarity measure of each cluster with its most similar cluster. Its minimum value is 0, with lower values indicating better cluster. Calinski Harabasz score is defined as the ratio between the intra-cluster dispersion and the inter-cluster dispersion. The higher the score, the more well defined the clusters are. Following is the comparison of the cluster analysis parameters for three different clustering algorithms:

S. No.	Algorithm	Silhouette score	Calinski Harabasz score	Davies Bouldin index
1.	k-means clustering	0.4992	13.624	0.5432
2.	BIRCH clustering	0.54483	34.2710	0.69887
3.	DBSCAN clustering	0.7250	87.7724	0.4296

The results of the performance metrics indicates that silhouette score of DBSCAN algorithm is best, as the value is much closer to 1 that indicates a good cluster. Calinski Harabasz score for a good cluster needs to be as high as possible. Higher silhouette score indicates the optimal number of clusters. As per the results obtained, DBSCAN algorithm has the highest calinski score. The davies bouldin index of the cluster needs to be close to zero for good quality of the cluster. Here again, DBSCAN has the lowest index value of 0.4 proving the best cluster quality.

6 Proposed System

6.1 System Architecture

Figure 6 depicts the overall architecture of cloud with integration of machine learning components. Cloud resource management plays a key role in resource provisioning. Due to cloud adoption at large scale, cloud resource providers need to take care of the scalability and dynamic requirements of the cloud consumer in real time. Traditional scheduling algorithms suffice the requirements of cloud resource scheduling and allocation. However,they fail to provide optimal resources when there is sudden burst of workload. In this scenario, there is a mere requirement of additional cloud components that can help the resource manager to allocate the optimal resources sufficing the consumer requirements as well as efficiently utilizing the cloud resources. Machine learning-based components in cloud can help in cloud resource allocation by grouping together the similar workload and assigning the resources accordingly. Here, we propose an ML-based system where the machine learning tools interact with resource manager and resource broker for efficient allocation of cloud resources. The proposed machine learning model has been built based on the earlier data of work-

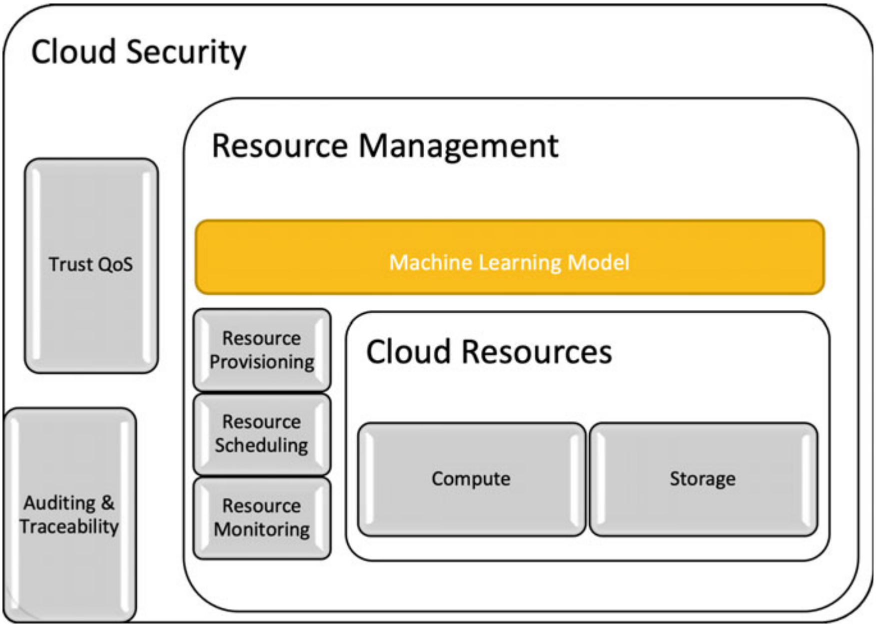


Fig. 6 System architecture for cloud resource management with ML-based tools

load generated by google cluster traces. The objective of the model is to generate the classes of similar workload into different clusters so that similar workloads are together in a single cluster. Using clustering-based model for workload classification provides two main advantages: It converts the unlabeled data into a labeled one and helps in dimension reduction of the input data.

6.2 Proposed Workflow

Figure 7 depicts the proposed workflow. When the user requests are received by the cloud resource broker, it will interact with the ML-based workload classifier, which will assign the job to a particular class based on the length of the job. The allocation of resources will be done based on the availability and workload mapping to cloud resources.

Following are the steps based on which the above built model will work:

1. User send the job requests to cloud service portal, jobids are assigned to them.
2. Cloud resource manager contact cloud broker to locate the best suitable resource to serve the request.
3. Cloud broker interacts with the clustering-based ML model to classify the job.
4. The classified workload is compared with the resource master database.

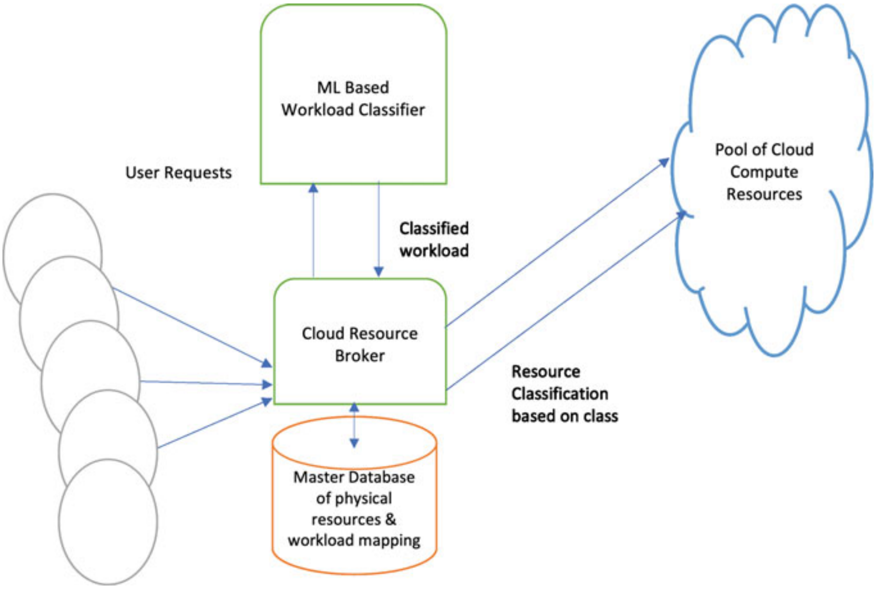

ML Based Workload Classifier
User Requests
Classified workload
Pool of Cloud Compute Resources
Cloud Resource Broker
Resource Classification based on class
Master Database of physical resources & workload mapping

Fig. 7 Proposed workflow

5. The class of each job is mapped with the physical resource category maintained in the database and each job is marked with the suitable cloud resource.
6. The job gets executed on the identified resource and output is sent to the end user.

7 Discussion

A lot of research has been done to work on the various aspects of cloud resource allocation including scheduling and workload balancing algorithms. Clustering techniques such as K-means clustering groups the similar data points together, however its very important to validate the quality of the clusters formed. This also indicates the quality of the ML-based model for workload classification and will indirectly effect the efficiency of the resource allocation mechanisms in cloud. The research work compares the quality of the clusters that are formed by various clustering algorithms on the cloud workload dataset. The proposed model has been suggested as a reference architecture and can be adopted for existing cloud frameworks. The validation of the model is planned to be conducted on Openstack-based cloud. The authors suggest the researchers working in cloud to utilize the DBSCAN-based clustering algorithm to group the incoming workloads based on the performance, metrics achieved in the above experiment.

8 Conclusions

Resource management and allocation in a distributed computing environment such as cloud computing is a challenging issue, as cloud data centers comprise heterogeneous resources. Also, the number of users and their requests vary dynamically in the cloud. Hence, there is a need for an intelligent system that can efficiently allocate the cloud resources as per the need of the task based on its length. Machine learning-based models learn based on the previous cloud workloads and train the model for classifying the tasks. This classification of tasks will help in predicting the optimized cloud resources for sufficing the workload requirement. Using unsupervised learning algorithms like K-Means Clustering and DBSCAN are very useful for classification of the incoming workloads on cloud. The clustering results obtained from above shows that the workload dataset has been segregated into different clusters in both K-means and DBSCAN. The accuracy of clustering in DBSCAN algorithm is dependent on the value of epsilon. Hence, an optimized value need to be chosen. For our dataset, 0.5 is the optimized epsilon value for which we are getting clear clustering of workload. The cluster quality obtained by the clustering algorithms has been analyzed using the clustering performance metrics like Silhouette score, Calinski score and Davies bould index. With the results obtained for performance metrics, DBSCAN clusters obtained are best of the three clustering algorithms. Hence, we conclude that using clustering-based ML approaches, we can effectively classify the incoming workload that will help the cloud resource broker for optimized resource allocation on cloud.

References

1. N. I. S. T. Nist (2015) definition of cloud computing. Technical report
2. Gonzalez N, Carvalho T (2017) Cloud resource management Miers. towards efficient execution of large-scale scientific applications and workflows on complex infrastructures. J Cloud Comp 6:13
3. Marinescu DC (2018) Chapter 9—cloud resource management and scheduling. In: Marinescu DC (ed) Cloud computing, 2nd edn, vol 9780. Morgan Kaufmann, pp 321–363
4. Feng M, Wang X, Zhang Y, Li J (2012) Multi-objective particle swarm optimization for resource allocation in cloud computing. In: 2012 IEEE 2nd international conference on cloud computing and intelligence systems, pp 1161–1165
5. Sarker IH (2021) Machine learning: algorithms, real-world applications, and research directions. Sn Comput SCI, 2:160
6. Yuan C, Yang H (2019) Research on k-value selection method of k-means clustering algorithm. Journal 2:226–235
7. Patel K, Sarje S, Vm KA (2012) provisioning policies to improve the profit of cloud infrastructure service providers. In: International conference on computing communication and networking technology (ICCCNT) IEEE-20180
8. Rathore S (2012) Efficient allocation of virtual machine in cloud computing environment. Int J Comput Sci Inform ISSN 5292(2):59–62
9. Shah M, Kariyani D, Agrawal L (2013) Allocation of virtual machines in cloud computing using load balancing algorithm. Int J Comput Sci Inf Technol Secur ISSN:9555(3):93–95

10. Panchal B, Kapoor R (2013) Dynamic vm allocation algorithm using clustering in cloud computing. Int J Adv Res Comput Sci Softw Eng 3(9):143–150
11. Ralba AH et al (2018) A computation-aware load balancing scheduler for cloud computing. Cluster Comput
12. Kanakardurga G, Veeramallu B (2014) Dynamically allocating the resource using virtual machines. Int J Comput Sci Inf Technol ISSN: 0, 9646(5):4646–4648
13. Bertolini M, Mezzogori D, Neroni M, Zammori F Machine learning for industrial applications: a comprehensive literature review, expert systems with applications 175:957–4174
14. Oyelade J et al (2019) Data clustering: algorithms and its applications. In: 2019 19th International conference on computational science and its applications (ICCSA, pp 71–81
15. Muthusamy G, Chandran SR (2021) Cluster-based task scheduling using k-means clustering for load balancing in cloud datacenters. J Internet Technol 22(1):121–130
16. Xiaowei JS, Martin X, Hans-Peter E, Kriegel (1996) Density based algorithm for discovering clusters. KDD-96 proceedings
17. Fontanini AD, Abreu J (2018) A data-driven birch clustering method for extracting typical load profiles for big data, 1–5
18. Google Research (2019) Google cluster workload traces. https://research.google/tools/datasets/google-cluster-workload-traces-2019/
19. Kavulya S, Tan J, Gandhi R, Narasimhan P (2010) Yahoo cluster traces: an analysis of traces from a production map reduce cluster. In: 2010 10th IEEE/ACM international conference on cluster. Cloud and Grid Computing, pp 94–103
20. Facebook (2010) Facebook hadoop workload. https://github.com/SWIMProjectUCB/SWIM/wiki/Workloads-repository
21. UC Berkeley AMP Lab (2010) Opencloud hadoop workload. http://ftp.pdl.cmu.edu/pub/datasets/hla/
22. University of California (2018) Eucalyptus IAAS cloud workload. https://www.cs.ucsb.edu/
23. Hussain A, Aleem M (2018) Google cloud jobs dataset for distributed and cloud computing infrastructures. MDPI Data J 3:38
24. Hussain MAA (2018) Gocj: Google cloud jobs dataset. https://data.mendeley.com/datasets/b7bp6xhrcd/1
25. Hussain A, Aleem M (2018) Gocj: Google cloud jobs dataset. Mendeley data, V, vol 1
26. Chormunge S, Jena S (2015) Metric based performance analysis of clustering algorithms for high dimensional data. In: 2015 fifth international conference on communication systems and network technologies, pp 1060–1064
27. Shahapure KR, Nicholas C (2020) Cluster quality analysis using silhouette score. In: 2020 IEEE 7th international conference on data science and advanced analytics (DSAA, pp 747–748
28. Liu Y, Li Z, Xiong H, Gao X, Wu J (2010) Understanding of internal clustering validation measures. In: 2010 IEEE international conference on data mining, pp 911–916

Big Data Analytics in E-Healthcare Using Hadoop and Hive

Richa Choudhary

Abstract New technologies like big data, cloud computing, machine learning could play a vital role in providing healthcare services to patients. The healthcare industry is producing a large volume of data which is increasing exponentially. Healthcare industry data is unstructured and is collected in different varieties. There is a dire need of processing these huge volumes of healthcare datasets to provide personalized treatment to the patients, to provide predictive analytics so the pre-diagnosis can take place. To predict the insights from existing patient data requires new tools and techniques as healthcare data is complex. Big data technologies such as Hadoop, MapReduce, Pig, Hive, and others provide the platform for healthcare data processing. Increasing rates of severe health diseases are impacting human life. One such kind of disease is cancer, this work presents the association between smoking and lung cancer. This paper presents the implementation of hive architecture for analyzing the lung cancer rate among active smokers dataset from centers of disease control and prevention government agency. The results obtained show that active smokers have a higher rate of lung cancer. It also addresses various challenges of implementing big data techniques over healthcare data.

Keywords E-healthcare · Big data · Hive architecture · Hadoop

1 Introduction

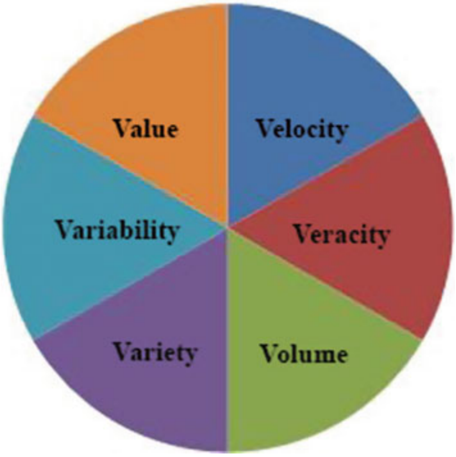
Big data is a huge collection of discrete raw data derived from various heterogeneous sources, which is difficult to process using current database management applications and traditional data processing tools. Big data may consist of various types of data such as text files, pdf files, audio, video, images, and data generated from sensor devices [1].

Big data is characterized by 6Vs as [2, 3] (Fig. 1).

R. Choudhary (✉)
School of Computer Science, University of Petroleum and Energy Studies, Dehradun, India
e-mail: richachoudhary.86@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_68

Fig. 1 Six versus of big data



Volume: The amount of data such as petabytes and zettabytes of data. This is about the huge amount of data being generated every second as a result of digitization in every domain [2, 5]. *Velocity*: It is the rate at which data is being generated. If we compare the data being generated today, it has increased exponentially in comparison with the previous few years. A massive amount of data is produced by every sector be it e-commerce, e-healthcare, banking, or any other sector [4, 5]. *Variety*: This represents various types of data: structured, unstructured, semi-structured key-value pair (KVP) datasets, etc. Data is in various formats like text files, audio, video, sensor-generated data, data from Facebook, Twitter, images sent from the satellite, online transaction data are some examples of types of data we are dealing with these days. To harness, such varied data can bring change to data analytics [2, 3]. *Value*: This represents the value that data can add. Data itself holds no value, but when the analytics are performed on the data to gather some insights, knowledge then the data is valuable. Each individual wants better satisfaction in terms of services and products, ease and availability, predictive analytics can help to ensure these to the end-users. [4]. *Variability*: It is the consistency in the data [5]. *Veracity*: It refers to the correctness of data. The primary aim is to ensure the trustworthiness of data [3]. A huge collection of data was already present, before the term big data. What is the differentiating factor between the big data era and before big data era? It is the role of an organization that wants to harness the data to gain insights and maximize its profit. The initiatives are taken by govt. in the field of big data also set apart the time before big data and the big data era. The focus is on the advancements and betterment of humankind that can be achieved through the analytics applied over big data. Generally, the term big data is confused with big datasets; it is a big misconception that big data means—huge datasets. Big data is far more than the size of the data, it is the collection of huge, varied, unstructured, and fast-growing data which is very large and highly complex to deal with, whereas if we are storing a huge amount of structured data and interrogating it with the help of SQL, then it is a large or big dataset. Big data requires distributed storage and memory analytics to extract useful information from a tremendous amount of unstructured data [5].

2 Big Data Analytics in E-healthcare:

Due to advancements in analytics techniques, various sectors are harnessing knowledge out of huge data they are storing over past years. The healthcare industry is also facing huge changes and volume in healthcare data. There are several large sources of data in healthcare as per JMIR Medical Informatics Review report [6] as genomics, EHR, medical monitoring devices, wearable video devices, and health-related mobile phone apps, and there are traditional medical files and doctor prescriptions and lab reports which are maintained by each hospital. The EHR is adopted by many countries now. EHR alone has generated more than 500 petabytes of data, and it is increasing exponentially [6]. Time requires to analyze this huge collection of data in the healthcare sector. The question here is why big data for healthcare data analytics?

The traditional databases are working fine as long as the data they have to analyze is clean and in a structured format. But if the data is not structured, incomplete, and can be in any format, then the legacy systems are failing to provide answers. SQL language needs to have a proper structure and completeness of data. The answer to all these problems is big data. Moreover, various wearable healthcare devices are generating a continuous feed of data. Storage and analyzes of such data require techniques like Hadoop, MapReduce, and some other techniques of big data [7]. To extract useful information, there is a need to deploy cost-effective methods to improvise people's health. Figure 2 portrays the potential impact of big data technologies on the improvement of the healthcare system. It shows the conversion of raw information into useful information in an optimized manner using big data technologies by addressing the following viewpoints: what happened, why it happened, what will happen, to detect how we can make it happen.

2.1 E-healthcare

To meet the rapid evolution pace of health information and communication technologies, there is a need to harness the technologic capabilities to optimize the analysis of fast-growing data at a higher speed and lower cost. ICT-based solutions are expected to contribute extensively to the future development of our health system [7]. In the present era, the novel source of power is not money in the hands of few, but the information in hands of many, to build healthy relationships with patients, there is a need to provide quality healthcare services that will lead toward convenience, control, and choice. The solution was provided by introducing the concept of e-health, which was brought into name light by industrialists while thinking of the concept of e-commerce

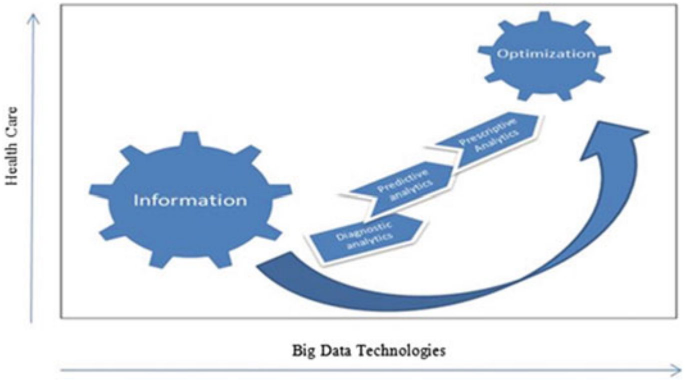
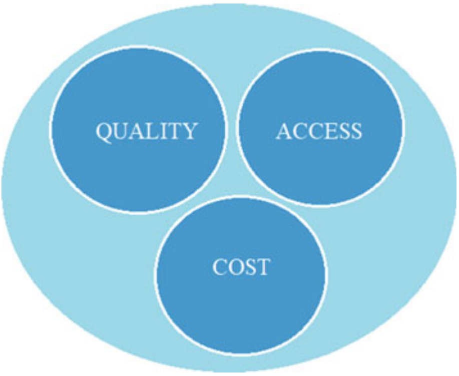


Fig. 2 Information optimization using big data technologies

and e-filling. E-health is one of the applications of ICT, which is broadly defined as, the use of health information and communication technologies (e.g., Internet, phones, electronic health records, electronic medical records, etc.) Assess, manage, monitor, and improve health services [7, 8] or in layman's terms, we can say e-health is the digitization of the whole healthcare system. E-healthcare services are driven by three pillar components known as iron triangle. As represented in the figure below quality, cost and access are the three competing healthcare issues that need to be focused on. For example, providing more affordable e-healthcare services will ultimately increase the access of e-health data and hence decreases the quality of healthcare [8]. To provide good quality e-health services, there is a need to increase the costs and limit access to it. So to provide better e-health services, these competing components need to be balanced (Fig. 3).

Apart from this as users of the Internet are increasing, a large percentage of Internet usage is dedicated toward health information. This statement is validated by the study conducted in 2005 which states "various e-health-related resources over

Fig. 3 Iron triangle of healthcare



Internet shows the same results as diagnosed by a doctor, or suggested the same treatment/drugs as prescribed by the doctors" [9]. As a result, a huge amount of e-health data is been generated and fetched. Hence, there is a need to analyze the generated data, to find some interesting patterns and useful information out of it. Impulsive growth in healthcare data has resulted in the enormous use of efficient, precise, and computationally effective computer-aided technologies. E-health researches and the information collected from such researches are complex to deal with, so there is a requirement to design and deploy the technology that can optimize e-health analysis in terms of speed and cost [10]. E-health applications, on one hand, have the potential to provide better healthcare services, but on the other hand, it also includes potential danger to patient privacy and health disparities by widening the digital divide. As a result, the e-health services will be confined to literate people. To bridge the gap between e-health services and its access to the people all around, some advent technology needs to be incorporated that will provide a balance among all the competing components of iron triangle to provide better and qualitative e-healthcare services [11, 12].

2.2 Importance of E-health

Traditional practices of the healthcare industry follow the paper and mail-based inefficient communication that surge the burden on patients and various healthcare providers and are prone to error and delay and waste resources, time, and energy. To share the health and patient information instantaneously and more easily among various health service providers (direct or indirect), there is a need to digitize the whole healthcare system or we can say there is a need to integrate ICT and e-business management in the field of healthcare. Many countries believe that it is important to increase public awareness about the emergency of moving the healthcare sector to a digital system for better performance, higher quality service [9, 11]. With the advantage, there comes some disadvantages and one of them is to unify the work or to run the work smoothly there must be some professionals in every healthcare center who can use the e-health system and ICT infrastructure for providing efficient and better services. To ensure the continued development of e-health initiatives, there is a need to build healthy partnerships with healthcare providers, local community organizations, and national healthcare associations [4, 5]. It means, there is a need to provide a platform where the above-mentioned partners can exchange their data and share their thoughts. The solution is using the Internet and communication technology in e-health. Or we can say development in the field of E-health is possible, by incorporation of fast and cost-efficient ICT-based techniques.

3 Challenges in E-Healthcare

Big data analytics has so much potential to harness healthcare data, but before that, some major challenges that need to be addressed. The major challenges of big data in e-healthcare are:

Security and Privacy of Data: Healthcare data security is crucial as privacy is the main concern for healthcare data. No one is interested in sharing the health issues they are facing. Healthcare organizations have to take utter care for handling such sensitive data. There is some govt. act as well to provide some guidelines while handling private data of patients, such as HIPAA. Seeing the sensitivity of healthcare data, security becomes the prime challenge [5, 6].

Data Standardization: No standard is fixed for healthcare data around the globe. There is no fixed technique that can deal with all sorts of data. The main sources of healthcare data are hospital medical records, prescriptions by the doctor, electronic health records (EHR), health monitoring devices, health-related apps. This gives a wide range of variety for the data. So, it is a challenge to access the entire repository of data of a single organization only. And across an organization, it is quite difficult to perform analytics because of the volume and variety of data [5, 8].

Storage and Retrieval of Data: Healthcare data is produced at high velocity these days, which is why it is difficult for traditional systems to store and process the data. The data available in the healthcare sector has so much variety that poses a challenge to store such data. Nowadays, due to advancements in technology, various wearable healthcare devices are available, which gives a continuous data stream to store [7]. It is a challenge to store such varied data and to make it available for analysis. Many healthcare organizations are going for cloud storage for healthcare data as these services are reliable and provide security to the data as well as transfers data from one location to another for big data analytics [7].

Capturing the Data: Capturing healthcare data and data ownership is another challenge. There should be some govt. initiatives to specify the rules for ownership of healthcare data as the healthcare organization may face some ethical issues while handling this sensitive data.

Lack of Technical Support: There is a big gap of data analysts available and required in the healthcare sector. In the current scenario, the available staff can analyze data using the traditional SQL queries, but to use advanced big data tools require technical expertise. Currently, there is a lack of support for using big data techniques. This poses another challenge in using big data for healthcare data analytics [13].

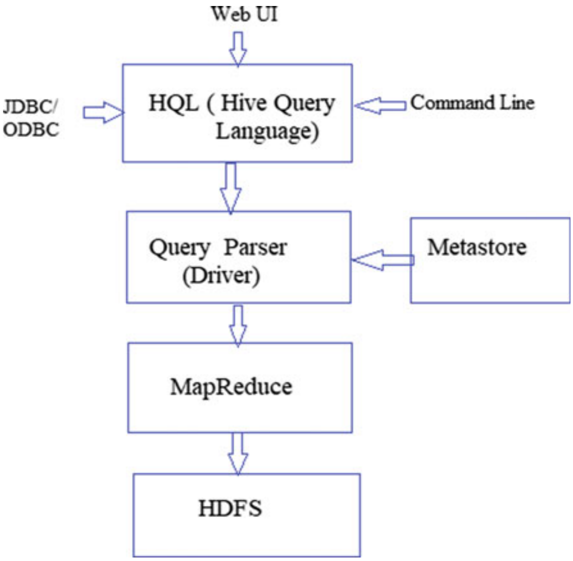
These issues of e-healthcare can be addressed using big data analytics. Big data technologies support any data format storage and processing. Many pieces of researches have been done which has produced data retrieval and visualization tools to help medical staff. These tools allows access to the patient records with an easy user interface.

4 Experiment

In this paper, we are presenting the big data analytics techniques on smoking patterns linked with CDC cancer across the USA using Hive and Hadoop. In the current era, many big data tools and techniques such as Hadoop, spark, NoSql, IMDb, blockchain, and R programming [11] have revolutionized medical data analytics and are providing accurate predictions and forecasting. Hadoop is a distributed file system to process a huge amount of data, and it includes other applications as well such as Hive, Apache-Spark, Pig, and many more. In this experiment, Hive is used to analyze the cancer rate based on smoking trends. Hive is built on the top of Hadoop to process SQL-like queries called Hive query language (HQL). Hive file structure is similar to the RDBMS, and it works with structured and unstructured data as it supports multiple formats. Hive can read data from multiple file formats. The data in Hive is processed in a batch of queries and the latency rate in hive processing can be medium to high, but it has a huge advantage in processing the same volume of data over RDBMS [14]. Figure 4 depicts the functioning of Hive to process the data:

Data processing in Hive is done using HQL, the queries can be provided to the system using a command line, Web interface, or the database connection established using JDBC for Java and ODBC for C++ . Queries submitted to the system are parsed using the driver to parse the queries. To parse the queries, the schema of the created tables in Hive is given by the metastore. Metastore stores the schema of all the objects created in HIVE. Parsed queries are submitted to MapReduce (MR) in a batch that further uses Hadoop distributed file system (HDFS) to process a large amount of data.

Fig. 4 Hive architecture



5 Results and Discussion

In this experiment, the hive is installed, configured, and loaded with a huge dataset. Hive objects are created for the data of smoking patterns. Here, we have shown the results analyzed using Hive technique of big data on smoking trend linked with centers for disease control and prevention (CDC) cancer data (Tables 1, 2, 3 and 4).

Table 1 Overall percentage change in the number of active smokers

YEAR	SAMPLE_SIZE	ACTIVE_SMOKERS	INACTIVE_SMOKERS
2011	503,925	104,482.83	399,442.16
2012	466,018	90,910.33	375,107.66
2013	476,451	89,787.36	386,663.63
2014	443,493	79,835.13	363,657.86
2015	423,367	73,142.93	350,224.06

Table 2 Percentage change in male in no. of active smokers

YEAR	SAMPLE_SIZE	ACTIVE_SMOKERS	INACTIVE_SMOKERS
2011	197,700	45,508.17	152,191.82
2012	187,756	40,497.12	147,258.87
2013	194,737	40,869.49	153,867.50
2014	184,020	36,488.17	147,531.82
2015	179,028	34,253.05	144,774.94

Table 3 Percentage change in female in no. of active smokers

YEAR	SAMPLE_SIZE	ACTIVE_SMOKERS	INACTIVE_SMOKERS
2011	306,225	56,626.57	249,598.42
2012	278,262	48,667.90	229,594.09
2013	281,714	47,246.64	234,467.35
2014	259,473	42,184.89	217,288.10
2015	244,339	37,840.78	206,498.21

Table 4 Percentage of smokers per state across the USA

STATE	STATE_ABBR	PERCENTAGE
West Virginia	WV	27.6561
Kentucky	KY	27.5082
Arkansas	AR	25.6328
Louisiana	LA	24.733
Mississippi	MS	24.6981
Tennessee	TN	24.1226
Oklahoma	OK	23.576
Indiana	IN	23.4814
Ohio	OH	23.1773
Missouri	MO	22.8542
Alabama	AL	22.7323
Michigan	MI	22.3247
South Carolina	SC	22.3246
Alaska	AK	21.428
Wyoming	WY	21.2743
Pennsylvania	PA	21.2338
North Dakota	ND	20.9602
South Dakota	SD	20.9171
Maine	ME	20.8865
North Carolina	NC	20.7089
Montana	MT	20.2428
Delaware	DE	20.2403
Kansas	KS	20.1345
New Mexico	NM	19.7837
Georgia	GA	19.6436
Nevada	NV	19.6205
Virginia	VA	19.5506
Wisconsin	WI	19.1963
Iowa	IA	19.135
District of Columbia	DC	18.9679
Nebraska	NE	18.9194
Illinois	IL	18.5579
Oregon	OR	18.0468
Minnesota	MN	17.99
Rhode Island	RI	17.82

(continued)

Table 4 (continued)

STATE	STATE_ABBR	PERCENTAGE
New Hampshire	NH	17.5624
Florida	FL	17.5237
Colorado	CO	17.3734
Vermont	VT	17.2217
Arizona	AZ	17.1779
Texas	TX	16.9073
Idaho	ID	16.6925
Washington	WA	16.6826
Massachusetts	MA	16.6474
Maryland	MD	16.4603
New York	NY	16.4268
New Jersey	NJ	16.3084
Connecticut	CT	15.9757
Hawaii	HI	14.7096
California	CA	12.9965
Puerto Rico	PR	12.4432
Utah	UT	10.5691

DATASET OBTAINED FROM - >

Behavioral risk factor surveillance system (BRFSS) age-adjusted prevalence Data (2011 to 2015)-> https://catalog.data.gov/dataset?tags=brfss

1. Create database journal;
2. Create a table in the hive to store data

```
create table smoke_trend(Year int,LocationAbbr
varchar(50),LocationDesc varchar(60),TopicType
varchar(100),TopicDesc varchar(200),MeasureDesc
varchar(80),DataSource varchar(15),Response varchar(60),Data_value_
Unit varchar(2),Data_Value_Type varchar(15),Data_Value
DOUBLE,Sample_Size int,Gender varchar(10),Race varchar(50),Age
varchar(30),Education varchar(50))
( row format delimited fields terminated by ’,’stored as TEXTFILE
tblproperties("skip.header.line.count"="1");
```

3. Storing data into table

```
Load data local inpath ’/home/shubham/Desktop/hive
projects/smoking_pattern/smoking_pattern_analysis.csv’ OVERWRITE
INTO table smoke_trend;
```

USE CASE -1: To find percentage change in the number of overall active smokers over 4 years [2011- > 2015]

```
selectYear,sum(Sample_Size)AS Sample_Size,sum((Sample_Size*Data_
value)/100) AS Active,sum((Sample_Size*(100.0-Data_value))/100) AS
Non_Active from smoke_trend > where (MeasureDesc =’Current Smoking’
AND Gender = ’Overall’ AND Race = ’All Races’ AND Age=’All Ages’) >
group by Year;
RESULT
```

II. MALE

```
select Year,sum(Sample_Size) AS Sample_Size,sum((Sample_Size*Data_
value)/100) AS Active,sum((Sample_Size*(100.0-Data_value))/100) AS
Non_Active from smoke_trend where (MeasureDesc =’Current Smoking’ AND
Gender = ’Male’ AND Race = ’All Races’ AND Age=’All Ages’)
group by Year;
```

III.Female

```
selectYear,sum(Sample_Size)AS Sample_Size,sum((Sample_Size*Data_
value)/100) AS Active,sum((Sample_Size*(100.0-Data_value))/100) AS
Non_Active from smoke_trend where (MeasureDesc =’Current Smoking’ AND
Gender = ’Female’ AND Race = ’All Races’ AND Age=’All Ages’)group by
Year;
```

USE CASE -2: To find the state with the maximum no. of active smokers

```
select e.LocationDesc,e.LocationAbbr,(e.Active/e.Sample_Size)*100
as Percentage from(select LocationDesc,LocationAbbr,sum(Sample_Size)
AS Sample_Size,sum((Sample_Size*Data_value)/100)ASActive,sum
((Sample_Size*(100.0-Data_value))/100) AS Non_Active from
smoke_trend where (MeasureDesc =’Current Smoking’ AND Gender =
’Overall’ AND Race = ’All Races’ AND Age=’All Ages’ AND Year<2015)
group by LocationDesc,LocationAbbr)e order by Percentage DESC;
```

```
LUNG CANCER DATA ->
https://www.cdc.gov/cancer/uscs/public-use/pdf/uscs-public-use-
database-fact-sheet-508.pdf
create table lung_cancer(State varchar(4),Range_xy varchar(10),Rate
Double)row format delimited fields terminated by ’,’stored as TEXTFILE
tblproperties("skip.header.line.count"="1");load data local inpath
’/home/shubham/Desktop/hiveprojects/smoking_pattern/lung_map_
incidence.csv’ OVERWRITE INTO table lung_cancer;
select State,Rate from lung_cancer order by Rate;
```

The result here shows the no. of cancer cases among the percentage of smokers across states in the USA. In Table 5, the highest percentage of active smokers is in West Virginia with 27.65%, whereas the highest rate of cancer among smokers is in Kentucky State with 91.4% of cancer cases among the smokers with 27.50%. The result also shows that a high percentage of smokers in an area is not directly linked with a high no. of cancer cases as shown for the West Virginia State. But a majority of the states show high the number of smokers and higher are the number of cancer cases (Fig. 5).

The results conclude that big data can efficiently analyze heterogeneous data with accuracy and can provide the statistics to optimize decision-making.

Table 5 Percentage of cancer cases across states in USA

State_Abbr	Rate of cancer
KY	91.4
WV	77.6
AR	77.4
MS	73.9
TN	73.8
ME	72.1
MO	71.9
RI	69.6
OK	69.5
LA	68.8
IN	68.6
NC	67.3
DE	67.3
OH	67
IL	65.6
AL	65
SC	64.2
MI	63.4
IA	63.3
PA	63.2
GA	63
MA	61.5
VT	61.3
NH	60.9
KS	60.1
CT	59.2
ND	58.4
WI	57.9
VA	57.5
NY	57.5
FL	57
NE	56.9
SD	56.3

(continued)

Table 5 (continued)

State_Abbr	Rate of cancer
MD	55.4
NJ	55.4
WA	55.4
AK	54.5
OR	53
MT	52.8
MN	52.3
TX	51.7
NV	51.7
DC	50.5
ID	48.8
AZ	47.1
WY	45.2
HI	43.7
CO	42.2
CA	42
NM	38.6
UT	25.6

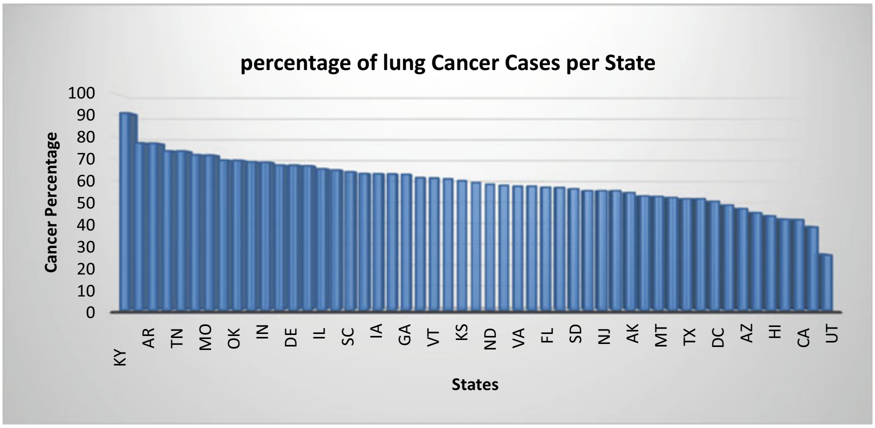


Fig. 5 Percentage of lung cancer cases per state in the USA concluded from smoking trends dataset

6 Conclusion

To build a strong healthcare infrastructure, key lies in digitizing this sector at a very fast pace. Delivering online access to specialized doctors, investigating the patients, and giving the first care online is crucial. Big data can help in providing better healthcare by providing the analysis of huge healthcare-related data on time. It can be used to find associations and patterns in the medical data. Here, the paper shows the linkage between smoking patterns and lung cancer rates in that region. These types of associations from the medical data can help to tackle severe diseases beforehand. In the current scenario, the biggest challenge in e-healthcare is to generalize the health data structure and to be able to store and process data received from heterogeneous sources. Big data tools and techniques have the infrastructure of storing and processing medical data and make it scalable as well. Hive processes and stores data in a distributed manner using HDFS and uses simple SQL-like queries. Big data techniques can make accurate predictions based on existing patient data. In future, these technologies can be integrated with artificial intelligence and the machine learning ecosystem.

References

1. Sagiroglu S, Sinanc D (2013) Big data: a review. Int Conf Collab Technol Syst (CTS) 2013:42–47. https://doi.org/10.1109/CTS.2013.6567202
2. https://www.ibm.com/in-en/analytics/hadoop/big-data-analytics. Last accessed on 27 July 2021
3. Tsoi K, Hung P, Poon S (2021) Introduction to the minitrack on big data on healthcare application. In: Proceedings of the 54th Hawaii international conference on system sciences, pp 3389)
4. Rehman A, Naz S, Razzak I (2021) Leveraging big data analytics in healthcare enhancement: trends, challenges and opportunities. Multimedia Syst. https://doi.org/10.1007/s00530-020-00736-8
5. Hariri RH, Fredericks EM, Bowers KM (2019) Uncertainty in big data analytics: survey, opportunities, and challenges. J Big Data 6:44. https://doi.org/10.1186/s40537-019-0206-3
6. Badawi O, Brennan T, Celi L, Feng M, Ghassemi M, Ippolito A, Johnson A, Mark R, Mayaud L, Moody G, Moses C, Naumann T, Nikore V, Pimentel M, Pollard T, Santos M, Stone D, Zimolzak A (2014) MIT critical data conference 2014 organizing committee. Making big data useful for health care: a summary of the inaugural MIT critical data conference. JMIR Med Inform 2(2):e22. https://medinform.jmir.org/2014/2/e22, https://doi.org/10.2196/medinform.3447
7. Chennamsetty H, Chalasani S, Riley D (2015) Predictive analytics on electronic health records (EHRs) using hadoop and hive. In: 2015 IEEE international conference on electrical, computer and communication technologies (ICECCT), pp 1–5. IEEE
8. Fangfang Lu, Chengzhong Xu, Pei Zhang, Yong Xu, Jianhua Liu (2021) Construction and implementation of big data in healthcare in Yichang City, Hubei Province[J]. China CDC Weekly 3(1):14–17. https://doi.org/10.46234/ccdcw2020.254
9. Kumar SR, Gayathri N, Muthuramalingam S, Balamurugan B, Ramesh C, Nallakaruppan MK (2019) Chapter 13—medical big data mining and processing in e-healthcare. In: Balas VE, Son LH, Jha S, Khari M, Kumar R (eds) Internet of Things in biomedical engineering, Academic

Press, pp 323–339, ISBN 9780128173565. https://doi.org/10.1016/B978-0-12-817356-5.00016-4

10. Bujnowska-Fedak MM (2015) Trends in the use of the Internet for health purposes in Poland. BMC Public Health 15:194. https://doi.org/10.1186/s12889-015-1473-3
11. Galetsi P, Katsaliaki K (2020) A review of the literature on big data analytics in healthcare. J Oper Res Soc 71(10):15111529. https://doi.org/10.1080/01605682.2019.1630328
12. Liu W, Park EK (2014) Big data as an e-health service. In: 2014 international conference on computing, networking and communications (ICNC), pp 982–988. https://doi.org/10.1109/ICCNC.2014.6785471
13. Belle A, Thiagarajan R, Soroushmehr SM, Navidi F, Beard DA, Najarian K (2015) Big data analytics in healthcare. BioMed Res Int
14. Kumar PS, Pranavi S (2017) Performance analysis of machine learning algorithms on diabetes dataset using big data analytics. In: 2017 international conference on Infocom technologies and unmanned systems (trends and future directions) (ICTUS), pp 508–513. https://doi.org/10.1109/ICTUS.2017.8286062

Communication Technologies, Security and Privacy

Blockchain Security: A Systematic Review

Parshwa Shah and Madhuri Chopade

Abstract Blockchain is a technology that is decentralized. It has the ability to tackle a wide range of industrial issues. A blockchain transaction's records are secured by cryptography, and each transaction is linked to previous transactions or records. Algorithms on the nodes validate blockchain transactions. As a final point, blockchains enable transparency, allowing each participant to keep track of transactions at any point in time. Smart contracts provide for safe transactions, reducing the risk of third-party interference. Readers will have a better understanding of how blockchain technology helps protect and manage today's users. There is a thorough report on diverse blockchain studies and security proposed by the research community, and their distinct implications on blockchain, in the review. This article concludes with a detailed description of the blockchain security followed by a discussion of the many varieties of security enhancements.

Keywords Blockchain · Security · Blockchain attacks

1 Introduction

The idea of a secure chain of blocks is certainly not a fresh one. In 1991, Stuart Haber [1] proposed a method for digitally timestamping electronic documents in order to prevent manipulating. However, in recent years, it has grown in prominence as a result of its application in blockchain technology to hold transaction records for the cryptocurrency "Bitcoin". Blockchain has the ability to "revolutionize apps and reshape the digital economy," according to experts [2]. By enabling collaboration without trust, blockchain holds immense promise for re-establishing "trust" in

P. Shah (✉) · M. Chopade
Gandhinagar Institute of Technology, Moti Bhoyan, Gandhinagar, India
e-mail: parshwas09@gmail.com

M. Chopade
e-mail: madhuri.chopade@git.org.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_69

society. Blockchain technology offers enormous promise for a wide range of applications and provides diverse foundations with a lot of flexibility. The technology aids resource management and ensures secure and effective communication. When using blockchain to conduct financial transactions between parties, trust is increased because it reduces the chances of fraud and automatically provides a record of movement. In the traditional framework, when it comes to monetary exchanges, people must faith in a third party to complete the transaction. However, blockchain will provide optimum security in exchanges. Each exchange should be recorded in a block, which will act as a record book. When an exchange is completed, a block is added to the blockchain, which serves as a permanent information database. When a block is finished, it is either added to another block or a new block is created. Every block in a blockchain has a hash of its previous block [3]. In its most fundamental form, blockchain is called a distributed ledger. Blockchain exchanges are nearly tamper-proof thanks to hashing and appropriate calculations. People may for the most part access historical transactions provided by a blockchain, yet changing historical transactions inside the record is somewhat inconceivable. This is expected to some degree to the way that it is scattered; however, it is additionally secured with different factors.

In Sect. 2, we started with a brief explanation of different types of blockchain followed by the inherent security features of blockchain in Sect. 3. The novelty of our paper started from Sect. 4, where we discussed various possible attacks to blockchain network and research done for counter measuring those attacks. Additionally, some security enhancements to blockchain has been discussed in Sect. 5 followed by some frameworks proposed by researchers in Sect. 6. In Sect. 7, we gave some topics on which future research can be carried out followed by conclusion and references at the end.

2 Different Type of Blockchains

There are three major kinds of blockchain technology.

2.1 Public Blockchain

From the security point of view, public blockchain is more secure as it is completely decentralized and no one is able to change past transactions; however, any node from the network is able to keep an eye on transaction, hence confidentiality is not maintained. Bitcoin and Ethereum are two examples.

2.2 Private Blockchains

In private blockchain, a centralized authority is assigned that means they can change or update any transactions, as a result becomes less secure. While, transactions are kept private in this blockchain. Proof of authority consensus is often used in private blockchain.

2.3 Consortium Blockchains

In consortium blockchains, a central authority preapproves known members before they may engage in consensus on a blockchain network. When using this "semi-permissioned" strategy, a network can be dispersed or partially decentralized, while yet maintaining some level of control. In banking or supply chain management, this form of blockchain is utilized between parties as to maintain security and save it from attacks.

3 Blockchain Features for Security

See (Table 1).

4 What Are the Effects of Security Attacks to Blockchain Network and Solutions Proposed to Combat that Attack

4.1 51% Vulnerability

Effect: There is a possibility of 51% launch on blockchains with proof of work (PoW) hashing control if a miner's hashing control exceeds half the entire blockchain. The content on blockchain might be deliberately altered by an intruder by launching a 51% assault. His control over that blockchain will be total.

Solution: Sayeed and Macro-Gisbert in their paper [9] tried focusing on cryptocurrencies that had low hashing power to demonstrate the flaws in the consensus process which bolsters this attack. They then provide 5 security techniques in their work. Another recent effort to combat this attack in named as "Permapoint" [10] minimizes the chain re-organization.

Table 1 Security features

Security feature	Definition
Ledger	It is an immutable database, in which transactions are only added. No one is able to delete or update past transactions. Data from ledger cannot be accessed by anyone [4]
Chain of block	Hash values are required for each block in blockchain. They are linked by their prior hash. If someone or an attacker changes some data then the hash of that block will change, and it will be unlinked to network which won't be possible. As a result, sensitive data or information will be better protected. It is impossible to proceed with a transaction if any of the nodes do not agree to it [5]
Smart contracts	Smart contracts are nothing but a computer program which acts as a lawyer between two transaction parties. A transaction can be carried out between two users only after they agree to smart contracts. Smart contracts, on the other hand, relate to scripts that are automatically performed on a shared database consisting of nodes that are mutually distrusting [6]
Consensus mechanism	At the point when a record update happens, an agreement cycle is used to approve exchanges and accomplish a concession to the exchange's effect. Consensus mechanisms known are: proof of work (PoW), proof of stake (PoS), proof of existence (PoE), proof of exercise (PoX), byzantine fault tolerance (BFT), proof of importance (PoI), proof of luck (PoL), and proof of elapsed time (PoET) [7]
Hash function	Corruption-resistance and one-way functioning are the hash function's fundamental criteria. In online or offline transactions, hashing is mostly used to protect the integrity of data. Using a hash function, you may verify the authenticity of a file obtained from an Internet source The usage of hashes in blockchain applications is becoming increasingly prevalent. Hash function SHA256 is the most widely used one in blockchains [8]

4.2 Selfish Mining Attack

Effect: By this attack, intruders can earn excessive incentives by wasting genuine miners' incentives. Forking a private chain is attempted by the attacker, who retains found blocks secretly. They would then mine on this secret chain and continue to achieve a considerably longer private branch than that of the public branch since they have more freshly found blocks on their own private chain. Fair miners are still working in public chains. So honest miners will waste computer resources and intruders will get incentives.

Solution: In order to mitigate this attack, the researchers tried using a genuine approach of mining to create truth state notation for each blocks along with allotting self-confirmation height to users' transactions.

4.3 Double Spending Attack

Effect: When some crypto assets are spent and those are then duplicated and spent again, then this process is called double spending attack. It becomes impossible to avoid double spending attacks. Example: 51% vulnerability, race, and vector76 attacks.

Solution: Nicolas and Wang introduced multistage secure pool which verify the transactions by using four well-defined steps. Begum et al. [11] present a series of countermeasures against double spending assaults.

4.4 BGP Hijacking Attack

Effect: At the point when packets are sent to their objective, border gateway protocol (BGP) is utilized as a routing protocol. Aggressors use BGP directing to capture the organization traffic of blockchain. To do BGP hijacking, network administrators should be in charge, which might be taken advantage of to postpone network traffic. A BGP attack on Bitcoin is investigated by Maria et al. [12].

Solution: A scheme named BGPCoin is proposed by Xang in [13] that creates smart contracts to conduct and manage allocation of resources on a temper-resistant Ethereum network. It is a reliable solution to this problem based on Ethereum and smart contract coding.

4.5 DAO Attack

Effect: A “decentralized and automated” smart contract allowed for duplicate withdrawals, putting people’s digital assets at risk. The “DAO” hack, for example, saw $60 million US dollars stolen from a “decentralized and automated” smart contract.

Solution: To combat this attack, researchers proposed a technique on trials conducted with a tool named Contiki (A low power built tool for resource constrained environment) [14].

4.6 Liveness Attack

Effect: Liveness attack is proposed by Aggelos et al. [15] in order to delay the confirmation time of a target transaction. Both Bitcoin and Ethereum have been attacked in two different ways. There are three steps to a liveness attack, namely assault preparation, transaction denial, and blockchain retarder.

Solution: Conflux's consensus protocol effectively encapsulates two distinct block generation algorithms developed by Li et al. [16] to prevent the active liveness attack. The first is the ideal method, which allows for speedy confirmation, while the second is the cautious technique, which ensures consensus advancement. It is scalable and distributed blockchain technology with maximum bandwidth and rapid verification. It combines these two methodologies into an integrated consensus process by employing a revolutionary adaptive weight mechanism.

4.7 Sybill Attack

Effect: Attackers fabricate their identity and enter in a peer-to-peer network in order to harm the reputation of the computer security system.

Solution: Swathi in her paper [17] presented strategy to combat this attack by observing other nodes' behavior and scanning the nodes that are only transmitting the blocks to a single user.

5 Solutions/Research Proposed for Better Security

Security is the primary emphasis when it comes to blockchain technology, which is continuously being discovered and enhanced in order to achieve the goal of giving customers with better sufficient protection.

5.1 Mixing

Mixing services were created to keep users' addresses separate. As a consequence, the observer's ownership of coins is obscured through mixing, which is essentially a random exchange of user's coins with other users' coins. These mixing services, however, do not offer security against currency theft.

Mixcoin

CoinJoin was the first mixing technique [18]. Bonneau et al. suggested Mixcoin in 2014 as a way to make anonymous payments in Bitcoin and Bitcoin-like coins. The first stage was the introduction of Mixcoin, a cryptocurrency that aims to minimize the risk of robbery by holding the mixing service "responsible" if it takes a customer's money. Mixcoin expands the anonymity set to enable all users to mix coins at the same time to protect against passive attackers.

TumbleBit

To solve Mixcoin's accountability and anonymity issues, TumbleBit [19] proposes a solution that is completely compatible with Bitcoin. TumbleBit enables parties to send money to each other via an untrustworthy Tumbler. During a TumbleBit era, no one, not even the Tumbler, can identify which payment paid which payee.

CoinShuffle

CoinShuffle [20] is a protocol that enables users to use Bitcoin anonymously. CoinShuffle is based on the dissent accountable anonymous group communication system and has many benefits over the Bitcoin mixing methods that came before it. It does not need the involvement of a third party (whether trusted, responsible, or untrustworthy), and it is fully compatible with the existing Bitcoin system.

5.2 Non-Interactive Zero-Knowledge Proof (NIZK)

Zcash, zk-SNARK

Zerocoins, on employ fixed-value coins, therefore the e-cash outcome could not support full-fledged nameless payments. Also, before payment, unnamed coins must be converted into nameless coins by someone else. Transactions, on the other hand, do not allow for the concealment of information or transaction amounts. It was thus recommended that we use a currency called Zerocash in order to solve these difficulties. Anonymity and data transaction privacy are particularly important features of Zerocash, which uses anonymous currencies. As a result of this, transactions using a coin are much smaller, and the verification duration is much shorter particularly less than six minutes.

Zero-Knowledge (Range) Proof

Making them unlikable is a common way to safeguard the secrecy and anonymity of a transaction. To complete the transaction, the electronic cash system must verify that the online payer has access to classified information, such as the address from where the cash is coming. Notably, the zero-knowledge proof was designed specifically for situations such as those described in the previous sentence.

5.3 Digital Signature

Hellman and Diffie created the notion of digital signature in 1976 when they invented public key cryptography [8]. In public key cryptography, digital signatures are used for source authentication, integrity, and non-repudiation [8]. Forgery is impossible with the digital signature algorithm (DSA). Some of the signature schemes are discussed below.

Group signature

This method [21] enables members of a group to characterize cluster signed communications anonymously. The following eight criteria must be met by the security components created by group signatures: dependability and integrity, no framing, unforgeability, traceability, anonymity, unlink ability, unforgeable tracing verification, and coalition resistance.

Aggregate signature

A typical digital signature method with an aggregation function based on co-GDH, and bilinear mapping is an aggregate signature [22]. When there are some different signatures on different messages from several users then all these signatures is summarized into one single signature. The burden of signature storage and verification is significantly reduced by the aggregate signatures.

Monero Ring Signature

It was initially based from CryptoNote to protect the source of certain transaction or user handling that transaction. Monero is a hybrid cryptographic model which protect users' anonymity as it utilizes ring signature technology. It is also worth noting that a collection of prospective signatories is put together to generate an individual signature that may be used for transaction authorization. Its security is so powerful that even in case of any dispute or theft, the original identity of user cannot be revealed.

Blind signature

The issue of big number factor decomposition, discrete logarithm problem, and elliptic curve is used to create a blind signature [23]. Its unique property is because it distinguishes message before it is signed. The main aim is to secure transmitter's privacy. Encrypted voting systems and digital currency schemes utilizes blind signatures.

Another digital signature technique is proxy signature [24].

6 Other Security Enhancements

There are some frameworks proposed by researchers in order to make blockchain network more secure and private. The following table provides details about the same. These concepts can be explained in detail but due to space constraint, they are summarized below (Table 2).

Table 2 Different types of countermeasures

Framework	Definition	Impact
Quantitative framework	It consists of a blockchain simulator and security model plan. The input parameters for consensus are network and protocol [25]	It generates a high-level fundamental process for detecting attacks
Oyente	It is mainly built to flaws in Ethereum contracts with the help of evaluating bytecode of contracts that are stored on Ethereum technology [26]	It is easy and simple to install on a system. All the bugs in Ethereum contracts are reported efficiently
Town crier	Town crier works as a mediator between clients and HTTP Web, as it gathers information from Web and then directs it to clients through blockchain network [27]	Information is secured will traveling as blockchain is used and also improves the reliability of transaction
Hawk	To improve security, developers use codeless smart contracts. This method increases privacy of smart contracts	The transactions on blockchain which are private are stored in a private part, while information which is not so important can be seen publicly [28]. It automatically generates cryptographic model in private smart contract
Lighting network	It uses double signing concept. A successful transaction is carried out only after both the parties involved signs the transaction receipts [27]	Third-party miner is not needed, which maintains confidentiality. Security is ensured due to double signing [29]
SegWit	It runs side by side, parallely to a blockchain network and signature data generated at the blockchain level is transferred to SegWit channel [30]	More blockchain space is liberated which results in faster transactions [31]. The signature data is stored in a Merkle tree. Network security has improved due to data diversity

7 Future Recommendations and Conclusion

There are various issues which are yet to be solved. Some of them are mentioned here:

- Firstly, many frameworks are there to mitigate attacks but a framework that can combat multiple attacks at the same time is a future research prospect in this field.
- Secondly, decentralized applications are increasing day by day and with that increases issues of data leakage. This problem should be solved using application hardening, code obfuscation, etc.

- Furthermore, at present, Bitcoin is used worldwide, and the use of cryptocurrencies at global level is increasing exponentially. This results in more criminal activities, with the help of cryptocurrencies, like money laundering, ransomware, and purchase of illegal goods like weed, cocaine, etc. For this, a friendly crypto architecture should be proposed which aids governments to find out those users who are performing suspicious illegal transactions to punish them accordingly.
- In future with the increasing use of quantum computing, traditional algorithms of digital signature can be easily decoded. For this, some researchers have suggested to use quantum cryptography. So, quantum-base key distribution requires more research.
- Consensus algorithms play a vital role in blockchain networks and prior research focused significantly on probabilistic reasoning. The difficulty of finding an efficient collection of parameters, modeling options, protocol variations, and compromises in the implementation of these algorithms is still unresolved.
- As private keys are an important feature, a framework for end-to-end communication of keys should be introduced.

This paper extensively discusses blockchain security and despite the fact that blockchain security is constantly improving, vulnerabilities continue to be discovered, and security research is ongoing. Furthermore, this study explored the many security difficulties, obstacles, and assaults that restrict the growing use of blockchain technology from a range of perspectives. For each assault, we discussed its effect and possible consequence. Eventually, we review recent advancements in blockchain security by different researchers and offered some recommendations for further research.

References

1. Anderson JR Security engineering: a guide to building dependable distributed sys
2. Singh S, Singh N (2016) Blockchain: future of financial and cyber security, in tems, 2nd ed. Indianapolis, IN, USA; Wiley, 2008. In: 2016 2nd international conference on contemporary computing and informatics IC3I, pp 463–467. https://doi.org/10.1109/IC3I.2016.7918009
3. Stephen R, Alex A (2018) In: IOP conference series: materials science and engineering 396: 012030
4. https://www.coindesk.com/information/who-created-ethereum
5. https://www.business2community.com/tech-gadgets/issues-blockchain-security-02003488
6. Bartoletti M, Pompianu L (2017) An empirical analysis of smart contracts: platforms, applications, and design patterns. In: Financial cryptography and data security: Springer, Cham
7. Abeyratne SA, Monfared RP (2016) Blockchain ready manufacturing supply chain using distributed ledger. Int J Res Eng Technol 5(9):1–10
8. Salman T, Zolanvari M, Erbad A, Jain R, Samaka M (2019) Security services using blockchains: a state of the art survey. IEEE Commun Surveys Tuts 21(1):858–880, 1st Quart
9. Sayeed S, Marco-Gisbert H (2019) Assessing blockchain consensus and security mechanisms against the 51% attack. Appl Sci 9(9):1788

10. Odera L (2020). Ethereum Classic and IOHK team up to find solutions to prevent 51% attacks on the blockchain. Accessed on 20 Dec 2020. [Online]. Available: https://bitcoinexchangeguide.com/ethereumclassic-iohk-team-up-to-find-solutions-to-prevent-51-attacks-on-the-blockchain/
11. Begum A, Tareq AH, Sultana M, Sohel MK, Rahman T, Sarwar AH (2020) Blockchain attacks, analysis and a model to solve double spending attack. Int J Mach Learn Comput 10(2):1–6
12. Apostolaki M, Zohar A, Vanbever L (2017) Hijacking bitcoin: routing attacks on cryptocurrencies. In: IEEE symposium on security and privacy, pp 375–392
13. Xing Q, Wang B, Wang X (2017) POSTER: BGPCoin: a trustworthy blockchain-based resource management solution for BGP security. In: 2017 proceedings ACM SIGSAC conference on computer and communications security, pp 2591–2593
14. Ghaleb B, Al-Dubai A, Ekonomou E, Qasem M, Romdhani I, Mackenzie L (2019) Addressing the DAO insider attack in RPL's Internet of Things networks. IEEE Commun Lett 23(1):68–71
15. Kiayias A, Panagiotakos G (2016) On trees, chains and fast transactions in the blockchain. URL https://eprint.iacr.org/2016/545.pdf
16. Chenxin L, Peilun L, Dong Z, Zhe Y, Ming W, Guang Y, Wei X, Fan L, Andrew CY (2020) A decentralized blockchain with high throughput and fast confirmation. In: Proceedings USENIX annual technical conference (USENIX ATC), pp 515–528
17. Swathi P, Modi C, Patel D (2019) Preventing sybil attack in blockchain using distributed behavior monitoring of miners. In: Proceedings 10th international conference on computing, communication and networking technologies (ICCCNT), pp 1–6
18. Meiklejohn M et al (2016) A fistful of bitcoins: characterizing payments among men with no names. Commun ACM 59(4):86–93. https://doi.org/10.1145/2896384
19. Heilman E, Alshenibr L, Baldimtsi F, Scafuro A, Goldberg S (2017) TumbleBit: an untrusted bitcoincompatible anonymous payment hub. In: Proceedings of NDSS https://doi.org/10.14722/ndss .2017.23086
20. Ruffing T, Moreno-Sanchez P, Kate A (2017) P2P mixing and unlinkable bitcoin transactions. In: Proceedings of NDSS. https://doi.org/10.14722/ndss .2017.23415
21. Chaum D, Heyst EV (1991) In: Proceedings of advances in cryptology—EUROCRYPT '91. Group Signatures (Springer, Berlin, Heidelberg, 1991), pp 257–265
22. Dan B, Gentry C, Lynn B, Shacham H (2003) In: Proceedings of international conference on the theory and applications of cryptographic techniques. Aggregate and verifiably encrypted signatures from bilinear maps (Springer, Berlin, Heidelberg, 2003), pp 416–432
23. Chaum D (1984) Blind signature system. In: Proceedings of advances in cryptology—CRYPTO '83, Santa Barbara, California, USA, August 21–24, pp 153
24. Mambo M, Usuda K, Okamoto E (1996) Proxy signatures for delegating signing operation. In: Proceedings of the 3rd ACM conference on computer and communications security. New Delhi, India, March 14–16, pp 48–57
25. Er-Rajy L, El Kiram My A, El Ghazouani M, Achbarou O (2017) Blockchain: bitcoin wallet cryptography security, challenges and countermeasures. J Internet Banking Commerce 22(3):1–29
26. Karame G, Androulaki E (2016) Bitcoin and blockchain security. Norwood, MA, USA: Artech House
27. Idelberger F, Governatori G, Riveret R, Sartor G, Evaluation of logic-based smart contracts for blockchain systems. In: Proceedings international symposium on rules and rule markup languages for the semantic web, Cham, Switzerland: Springer, pp 167–183
28. Aitzhan NZ, Svetinovic D (2018) Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams. IEEE Trans Depend Sec Comput 15(5):840–852
29. Zaghloul E, Li T, Mutka MW, Ren J (2020) Bitcoin and blockchain: security and privacy. IEEE Internet Things J 7(10):10288–10313. https://doi.org/10.1109/JIOT.2020.3004273
30. Kiayias A, Panagiotakos G (2015) Speed-security tradeoffs in blockchain protocols. IACR Cryptol ePrint Arch 2015:1–27

31. Cachin C (2017) Blockchains and consensus protocols: snake oil warning. In: Proceedings 13th European dependable computing conference (EDCC), pp 1–2
32. https://www.dotmagazine.online/issues/innovation-in-digital-commerce/what-can-blockchain-do/securityand-privacy-in-blockchain-environments
33. Natoli C, Gramoli V (2016) The balance attack against proof-of-work blockchains: the r3 testbed as an example. ArXiv preprint: arXiv:org/1612.09426
34. Rivest RL, Shamir A, Tauman Y [n. d.] How to leak a secret, pp 552–565

Adoption of Blockchain Technology to Secure Electronic Healthcare Record for Efficient Healthcare System

Aman Velani, Tirth Shukla, and Mitul Maniar

Abstract Today, blockchain technology has become one of the most important inventions and unique breakthroughs that have had a significant impact on people and various sectors. Specifically, in the healthcare sector, blockchain shows a promising future in storing medical records on the internet in a secure manner. Through this paper, we are spreading light on the Electronic Health Records (EHR) system, which is based on blockchain technology. As we all know, a blockchain ledger is an immutable, secure, and organized database. Thus, storing medical records of individuals on this ledger will help them to efficiently share documents with their authorized doctors, cloud storage facilities, as well as tamper-proof hash cryptography of data. There are plenty of different architectural approaches to attain this proposed objective. But, the distribution of public keys and private keys is the core system for EHR as it is developed on blockchain. The public and private keys are distributed in EHR to different clients for accessing data, along with that user's ability to change the rights accordingly from view to edit. Blockchain-based Personal Health Records (PHRs) systems have emerged as the newest thing in patient-driven healthcare.

Keywords Blockchain · Healthcare · Electronic health records · Limitations · Survey

A. Velani (✉) · T. Shukla · M. Maniar
Gandhinagar Institute of Technology, Gandhinagar, India
e-mail: amanvelani@gmail.com

T. Shukla
e-mail: tirthshukla13@gmail.com

M. Maniar
e-mail: mitul.maniar@git.org.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_70

1 Introduction

A blockchain exchange in the public record contains an undeniable record and when the data is entered, it can't be changed or eradicated later on. Blockchain innovation disposes of outsider go-between and takes into consideration confirmation and exchanges straightforwardly. Single or several records cannot be altered without modifying every block on the blockchain. The records on a blockchain fill in as publicly accessible digital records.

Hospital services have always actively worked to decrease the patient's costs by making use of medical information data. But this has unfortunately brought some new challenges to the medical institutions. Electronic Health Records is electronic personal medical health record that contains all the personal health-related data like medical images, medications, reports, genetic disease, etc. and this could help patients as well as hospitals to increase the cure rate. In the current system implemented centralized cloud servers are set up on the cloud to store the data. Progress in innovation has the greatest potential to help enhance security, user experience, and other elements of the medical services sector. Electronic Health Record (EHR) framework offered several benefits. However, they have to deal with several concerns, such as the security of clinical data, client accountability, and the reliability of the information, to name a few. The EHR framework additionally deals with certain different issues which are interoperability, data unevenness, and information breaks. A survey has shown that by 2020, the amount of medical data will double in the upcoming years. Most of the data surrounding healthcare are in an unstructured format. Millions of dollars are wasted on containing this unstructured data. Maintaining the system also requires a lot of resources.

Blockchain might be the solution to these problems. EHRs comprise fundamental and very sensitive private information that is used in medical services for diagnosis and treatment. This data is a valuable source of information on medical services. In an EHR, a patient's wellbeing information is established and sustained throughout a person's life, and it is often stored by and shared with a variety of medical clinics, health care institutions, and wellness providers [1]. Blockchain innovation is reclassifying information displaying and administration conveyed in numerous medical services applications. This is chiefly because of its flexibility and capacities to fragment, secure, and share clinical information and administrations remarkably.

This paper focuses on the limitations of traditional systems and the impact that blockchain technology can have on the Electronic Health Records (EHR). The limitations are also mentioned that blockchain technology has to overcome to have a private and efficient way to store and share health records that could help cure patients faster.

2 Traditional EHR Exploration

To understand the current scenario of the usability of the EHR we would like to give an example that is a possible real-life scenario. Let’s say, two patients require major clinical treatment. The patients show up at the physician with different symptoms, for example, loss of breath, blockage, or headache, and are examined thoroughly before the treatment starts.

If the patient says that he/she has a main care doctor meaning the doctor who usually sees the patient and has a medical history of the patient but it so happens that physicians clinic doesn’t use a comparable medical care framework, therefore, the clinic has to contact the primary doctor for the patients EHR. The physician has to log in to an external medical care framework compatible with the one used by the primary doctor. After examining the data, the clinic has a clearer understanding of the medical history of the patient.

Now it is important to note that this could not have been possible if the patient does not inform the primary doctor to share the data about the patient’s previous healthcare visits. After access to the previous records is provided, the physician can carry out a careful examination of the data and after careful examination, he/she can recommend another specialist. Now the same thing will happen at the specialist’s clinic as they also don’t have access to the previous data and this cycle repeats itself.

Subsequently, to make requests to the primary doctors to obtain health data, it also takes a few days to weeks for the information which might be unexpected and deficient. And another scenario could be that the patient has some data regarding his health stored someplace else not able to be obtained by the doctor.

In these situations, it would be really helpful to have a blockchain registry where the patient’s information is kept, the doctor could then easily identify where the patient was seen at which other facilities or any emergency healthcare center. They may likewise check if the patient has received medication for the identical problems in some other emergency clinic. If the clinic has all this information then they could have much more clarity on the health of the patient, and previous records also which will help them to analyze and give treatment accordingly. This would result in saving of time, expenses, increased efficiency, and reliability. These abilities are what offer blockchain the chance to substantially influence the productivity and costs of medical care transportation. Central problems encountered in medical services involve lack of administration, and also how information may be made verifiable, permanent, and unchangeable. Blockchain innovation may be used to offer data set administrations for automated database applications for gathered and protected data [2].

3 Blockchain in EHR

Medical services as an industry have interesting prerequisites related to security and privacy because of extra legitimate necessities to ensure patients’ clinical data.

As wellbeing data is turning out to be more effective to acquire through shrewd gadgets, and patients are venturing out to numerous specialists, the sharing and privacy of this data are a worry [3]. The critical objectives in the execution of a secure blockchain-based framework in medical care are as follows:

- Security: availability, confidentiality, and integrity.
- Auditability: it is a crucial concept of protection. Like, keeping a record of who is accessible for which data in PHR and EHR, the purpose behind it, and at what it was accessed.
- Authenticity: the potential to verify requestor identity by granting access to sensitive information.
- Privacy: confidentially handling private information, and only legitimate users will have access to the information provided.
- Anonymity: for the sake of secrecy, individuals have no public identification.
- Accountability: a person or a company would be investigated but also held accountable for their actions.

To fulfill the above-mentioned objective, there are some features included in the blockchain for healthcare that is discussed in the following Table 1 [4].

3.1 System Architecture

The one-of-a-kind necessities the healthcare industry is confronting. In light of the necessity of another form of secure EHR systems and the qualities of blockchain are executed. Various block transaction techniques and arrangements are shown in the structure. In the proposed system of blockchain architecture, there is a distribution of public and private keys in EHR to different clients.

Within the suggested framework, there are three members, which show restraint, the Admin system, the therapist, and the laboratory. The testament administrator then offers the approval as well as a secret key with some other ID to choose the user. The blockchain with hyper ledger fabric is used to distribute all trades. Groups have different roles in the network and can only view information to which they have been given access. The customer application is used by clients to add entries, which generates the chain code needed to send a transaction to the network. Following the transmission of the trading to the blockchain network, final actions are distributed across the network, guaranteeing each interaction is transmitted to each user of the architecture and that no trading may be changed or deleted by unauthorized consumers. The overall infrastructure is protected as transactions are normally attached to the former hash together with a timestamp. The records are modernized as well as visible to all clients in the blockchain network. Providers, including physicians as well as research unit personnel, will use the system to get the content they need. The physician or laboratory worker can examine and reload the permission information of the patients at any time, but he/she must have the authority of patients to carry out this work [2] (Fig. 1).

Table 1 Trait of blockchain in EHR

Identity manager	Within the network, the authenticity of every participant's identification must be ensured. To guarantee security measures and prevent cyberattacks, only genuine requests are accepted
Data sharing	In many of these current medical systems, network operators are often the principal data custodians. With the right to self in mind, it is becoming increasingly popular to restore ownership of medical collected data, who is free to share his private details as he sees fit. It's also important to establish safe data exchange between companies and sectors
Data storage	For storing a variety of confidential medical data, a database with a trustworthy ledger named blockchain is used. When secure storage is established, anonymity should be assured. In reality, nevertheless, the amount of health records is typically vast and varied. As a result, a similar issue is figuring out how to cope with massive storage devices without harming the efficiency of blockchain architecture
Data audit	When disagreements emerge, audit logs can be used to hold recipients responsible for their dealings with EHRs Several applications use smart contracts along with blockchain to keep track of transactions for interoperability. Every action or query will be logged on the blockchain ledger and accessible at whatever moment
Pros.	• Better exchange of medical records among patients and doctors • High level of security • Confirm the legitimacy of billing management • The supply chain in medical is strengthened
Cons.	• Lack of willingness to accept this technology among people • Scalability • Hard to come up with rules and regulations as a different nation has different laws for the management of Health records

3.2 Application in Insurance Claims

Application in protection claims with the quick improvement of the clinical protection business, the quantity of guaranteed people has risen drastically, and protection guarantee occasions have likewise expanded. And yet, we can without much of a stretch discover that during the genuine protection claims, there are many instances of clinical protection misrepresentation through altering the clinical records, there are additionally many instances of denying claims because the clinical records are deficient. One might say that the clinical record is a significant reason for the insurance agency to pay for the clinical costs while reviewing the cases. Notwithstanding, the customary case the executive's framework has a few issues, and the legitimacy of the case is compromised. For instance, after an arrangement has been reached among patients and some individuals from the emergency clinic, they fashion the patient's clinical record and afterward wrongfully get a guarantee. Because of the above issues, we should guarantee the practicality, fulfillment, and particularly credibility of clinical records.

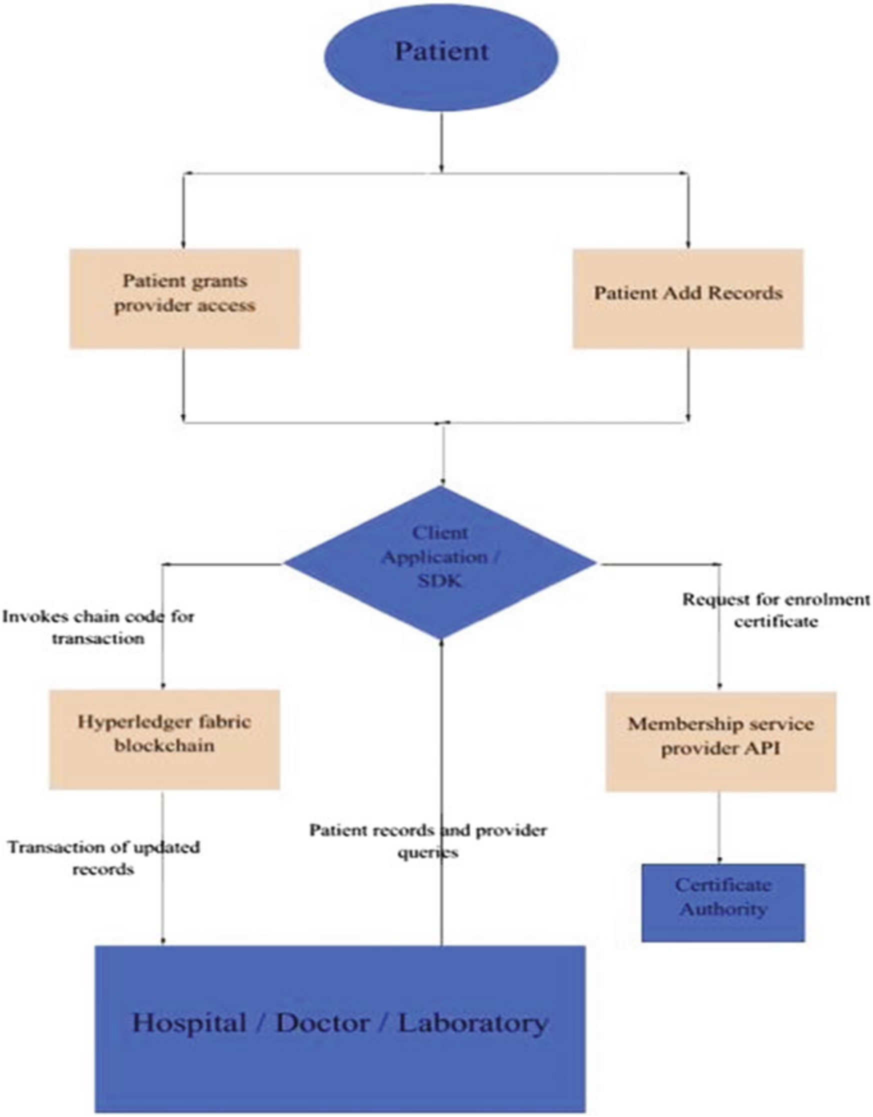

Patient
Patient grants provider access
Patient Add Records
Client Application / SDK
Invokes chain code for transaction
Request for enrolment certificate
Hyperledger fabric blockchain
Membership service provider API
Transaction of updated records
Patient records and provider queries
Certificate Authority
Hospital / Doctor / Laboratory

Fig. 1 Flow chart system architecture

Our protected EHR system can give a decent arrangement. Just the real clinical records can ensure the straightforwardness of the insurance claims measure and offer better assistance for insurance claims. In this system, as the clinical record data is created, the blockchain records the clinical record data whenever, and the clinical record data on the blockchain can't be adjusted subsequently. The agreement node constrained by the insurance organization in the agreement network gets data continuously. Accordingly, this likewise guarantees the integrity and realness of clinical records. At the hour of the case, the insurance organization can access the encrypted clinical record data to finish the case. We give a model for the clinical insurance scene. There are three sorts of substances in this cycle that are patients, medical clinics, and insurance organizations.

Primarily, hospitals, as well as insurance companies, get the authority to access the records, which are granted by the patient. Secondly, hospitals must submit the record with full encryption as per the information granted policy to the blockchain pool. Eventually, the source of medical data is cross-checked with the nodes which are consensus in blockchain, at the same time, a gathering of encrypted data is done on the cloud. When insurance companies get the implementation, they look for the availability of records and obtain the location of that data on the cloud. Afterward, the company can download that file and decrypt it with the private key.

In this system, the decentralization of the blockchain guarantees that clinical records are finished by different clinical foundations, accordingly staying away from the chance of a solitary establishment being controlled or paid off to record the records. Our framework can likewise uphold brilliant agreements for programmed claims settlement. At the point when the patient arrives at the case conditions, the patient will naturally get remuneration from the fund pool. This cycle doesn't need the presence of a customary insurance organization [5].

4 Limitations

Blockchain initiatives have gained a lot of attention in recent years. On the grounds that blockchain is a unique and complicated invention, it really would be erroneous to believe that we are managing innovation with the immediate application or that modifications can be implemented efficiently. We're still in an exploring phase. As a result, the blockchain won't be able to address all that is wrong with the current administration of health information. In any case, it offers a few opportunities for further developing the framework we have today, and that is the motivation behind why it is so fascinating and testing to investigate [6]. A patient-centered PHR system based on blockchain technology has many obstacles, which we highlight and explain [7].

4.1 Scalability and Performance

While it is possible to envisage storing all EHRs in a blockchain, large clinical records (such as X-rays and ECGs) are too large for direct storage. Considering [8] and [9]'s examinations of this problem, it remains a difficulty. As a result of blockchain implementation characteristics like decentralization, provenance, and consensus, all blocks should be kept on each user node on the system. To achieve scalability, demand on each node will rise as the size of information grows. According to the end of March 2018, a miner's complete involvement in Bitcoin's network required him to download all the data from the Bitcoin ledger, which came to more than 184 GB [10]. In addition, the Bitcoin network's most severe exchange approval is seven transactions each second, which increases the likelihood of a presentation bottleneck [9]. At this point, a blockchain-based stage that can contain substantially larger amounts of data will need proof of concept [11]. This problem was addressed in [8] by proposing to keep a large amount of clinical data off-chain in an information archive termed: *data lake". If the access control policy is enforced by the blockchain layer, this will be safe. In this structure, "the patient would, in any case, have control of who has access to the individual data in the data lake because the data would not be readable without the decryption key, which is stored on the patient's blockchain account" [8].

4.2 Usability

The great majority of people will be unfamiliar with the cryptography concepts behind blockchain transactions. Patients are expected to deal with their key sets (private or public) to offer digital fingerprints and approve authorization to their personal clinical information, according to the stated plans of the planned research. However, the difficulty of handling keys should be hidden behind online and/or phone applications with easy-to-use user interfaces [8] But it also creates a security risk. Self-administration represents another test if the patient can't approve fundamental access permits. An urgent critical sickness such as Alzheimer's disease or the loss of personal keys may be the cause. Medical personnel may also need to access the patient's data in an emergency if the patient is severely harmed.

MIT Media Lab inspected digital certificates carried out with blockchain innovation. Lessons learned from its early testing include the following: "it is substantially harder to oversee public/private keys to verify both issuer and recipient, thus building up a wallet that keeps certificates; as Bitcoin holds cash, perhaps another option" [9].

4.3 Secure Identification

Identifying patients' health data across multiple healthcare supplier backends (labs, HIEs, EHRs, medical clinics) is essential and non-trivial in the healthcare industry. In the United States, the Centers for Medicare and Medicaid Services (CMS) has set a significantly higher spotlight on medical services interoperability through its "Advancing Interoperability Program," which means to make patient data more available to partners. The same interoperability concerns are being addressed by new firms. As they are positive, these innovations and adjustments in approach do not answer the question of who has access to this patient information in the first place. Who or what is the real goal? It's all about identification, and in reality, identity management is a key component of blockchain technologies. We may connect a client's gadget (e.g., mobile phone) to a unique and crypto-secure payment system using a range of related advances. The final step in completing this innovation is to be certain that it was XYZ's cell phone that just drained tokens from XYZ's digital wallet. Cell phones and digital wallets, on the other hand, are not individuals. As intermediates in the best-case situation, they're prone to failing, getting stolen, and occasionally just getting lost. Adding unobtrusive biometrics on top of blockchain that doesn't compromise security guidelines could be a start to better describing the impact of the unknown, uninsured patients on medical costs [10].

4.4 Lack of Incentives and Willingness to Adapt

Building a big network of interconnected nodes is a major financial undertaking. For instance, as of late EHR frameworks were built and cost a huge number of dollars, and as of late, numerous huge health frameworks, boosted by the legislatures around the world, put resources into building business EHR frameworks [11]. Requesting a computerized version of the current record framework is by all accounts flippant spending for the tax-paying citizens and will be an injury to the clinical field. A sufficient number of nodes must be up and running at all times so that the consensus mechanism can function correctly and LO can give a minimal amount of valid signatures. Instead, blockchain would play a more supplementary role and not completely replace the current frameworks to progress the current situation. There would be a limited amount of descriptive international data stored in each node regarding a particular patient's information or conducted operation, but all of the pathology results would be kept off of the blockchain.

4.5 Interoperability

A worldwide solution would require numerous installations of the smart contracts to integrate, which would be difficult to do. Patients who travel to another nation must register anew under the Controller smart contract of the new jurisdiction [7].

4.6 GDPR

Due to blockchain's immutable nature, all information stored on-chain cannot be retrieved. Meaning if a patient wants to delete the data he/she can't do it. It would remain on the blockchain forever [7].

4.7 Smart Contracts Upgradability

Ethereum blockchain allows smart contracts to be kept on-chain and hence unchangeable. This, however, creates a serious problem, as their immutability prevents them from being upgraded. They cannot be changed after they have been developed and deployed. As a result, it is impossible to believe that a software update would be able to cure security vulnerabilities or software problems [7].

5 Conclusion

Blockchain in EHR is thoroughly analyzed, in this review paper, the motto behind this is to identify along with discussing the major benefit, disputes, and problems from blockchain in the healthcare industry. In the economic sector, this technology has made many advances. After examining, it is deduced that in the future the answer to major healthcare problems can be given with the help of blockchain. For instance, generating faith in the exchange of records between medical institutes, EHR, privacy, authorizing the laboratory or companies to access records and audibility. Nonetheless, it is crucial to carry out proper research in this field, before implementing it, as it has personal records and whole patients' history.

References

1. Mayer AH, da Costa CA, da Righi R, R. (2019) Electronic health records in a blockchain: a systematic review. Health Inf J 26(2):1273–1288. https://doi.org/10.1177/1460458219866350
2. Tanwar S, Parekh K, Evans R (2020) Blockchain-based electronic healthcare record system for healthcare 4.0 applications. J Inf Secur Appl 50:102407. https://doi.org/10.1016/j.jisa.2019.102407
3. McGhin T, Choo K-KR, Liu CZ, He D (2019) Blockchain in healthcare applications: research challenges and opportunities. J Netw Comput Appl 135:62–75. https://doi.org/10.1016/j.jnca.2019.02.027
4. Shi S, He D, Li L, Kumar N, Khan MK, Choo K-KR (2020) Applications of blockchain in ensuring the security and privacy of electronic health record systems: a survey. Comput Secur 97:101966. https://doi.org/10.1016/j.cose.2020.101966
5. Wang H, Song Y (2018) Secure cloud-based EHR system using attribute-based cryptosystem and blockchain. J Med Syst 42(8). https://doi.org/10.1007/s10916-018-0994-6
6. Capece G, Lorenzi F (2020) Blockchain and healthcare: opportunities and prospects for the EHR. Sustainability 12(22):9693. https://doi.org/10.3390/su12229693
7. Madine MM, Battah AA, Yaqoob I, Salah K, Jayaraman R, Al-Hammadi Y, Pesic S, Ellahham S (2020) Blockchain for giving patients control over their medical records. IEEE Access 8:193102–193115. https://doi.org/10.1109/access.2020.3032553
8. Cichosz S, Stausholm MN, Kronborg T, Vestergaard P, Hejlesen (2018) How to use blockchain for diabetes health care data and access management: an operational concept. J Diab Sci Technol 13(2)
9. Angraal S, Krumholz HM, Schulz (2017) Blockchain technology: applications in health care. Circ Cardiovasc Qual Outcomes 10(9)

10. Kassab M, DeFranco J, Malas T, Neto VVG, Destefanis G (2019) Blockchain: a panacea for electronic health records? In: 2019 IEEE/ACM 1st international workshop on software engineering for healthcare (SEH). 2019 IEEE/ACM 1st international workshop on software engineering for healthcare (SEH). https://doi.org/10.1109/seh.2019.00011
11. Griggs KN, Ossipova O, Kohlios CPC, Baccarini AN, Howson EA, Hayajnch T (2018) Healthcare blockchain system using smart contracts for secure automated remote patient monitoring. J Med Syst 42(7)

Secure Data Sharing in Medical Cyber-Physical system—a Review

Atharva Sarode, Komal Karkhile, Sanskruti Raskar, and Rachana Yogesh Patil

Abstract With the increase in advanced technologies, the health industry has developed a lot recently. Medical cyber-physical system (MCPS) plays a vital role in this. It consists of various medical devices which are networked together for smooth and efficient working. The patient's EHR is collected and stored on the cloud which is then easily accessible by doctors. The health industry has always been on top of cyber-attacks. With the onset of the Covid-19 pandemic, there was a sudden surge in telemedicine adoption, remote working and makeshift sites for virus testing and treatment, and under-preparedness, all contributing to new vulnerabilities and giving cybercriminals a new opportunity. In the medical world, we need our medical systems to be secure, reliable, efficient, and should ensure economic data storage and sharing for both patients and medical institutes. Motivated by these facts, we did a review on various MCPS techniques and algorithms that already existed. In this paper, we tried to summarize those techniques and provide a comparative study for the same.

Keywords MCPS · Cyber-attacks · Encryption · Blockchain · Cloud computing introduction

1 Introduction

Cyber-physical systems also abbreviated as CPSs are nothing but interconnection between the real and virtual world or [24] also can be said as a computer system in which a mechanism is controlled or monitored by computer-based algorithms.

A. Sarode (✉) · K. Karkhile · S. Raskar · R. Y. Patil
Pimpri Chinchwad College of Engineering, Pune, Maharashtra, India
e-mail: atharvasarode0@gmail.com

K. Karkhile
e-mail: komal.karkhile19@pccoepune.org

R. Y. Patil
e-mail: rachana.patil@pccoepune.org

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_71

CPS is currently very useful. They are used under different domains such as military, transportation, and health care.

The cyber-physical systems which are used in the healthcare domain are known as medical cyber-physical systems. This consists of various medical devices that are collectively involved in treating the patient [9]. The real and virtual interconnection here can be explained as various devices like pacemakers, medical ventilators, and infusion pumps are connected to the patient using various sensors and actuators, etc., and the data collected by these devices are stored using various encryption algorithms, cloud computing, blockchain, etc. But nowadays, MCPS does not deal only with connected devices but also involves big-data systems used in hospitals to attain unbroken and improved healthcare facilities. These medical systems keep a continuous record of the patient's state. The amount of data produced also known as electronic health records (EHRs) is on a large scale and plays a vital role in better treatment. Taking into consideration the amount of data produced, local devices cannot be used for such kind of data storage. Cloud computing, blockchain, etc., can be a few ways for effective data storage. Figure 1 shows us the simplified architecture of the medical CPS [34]. The data thus stored are further used by various hospitals in order if the patient requires any international consultation, it can be shared effectively. Even the data can be shared for medical insurance purposes on the patient requirement. It can also be used for disease analysis to understand new strains, symptoms of any newly arrived disease to recover and reduce the spread of disease instantly.

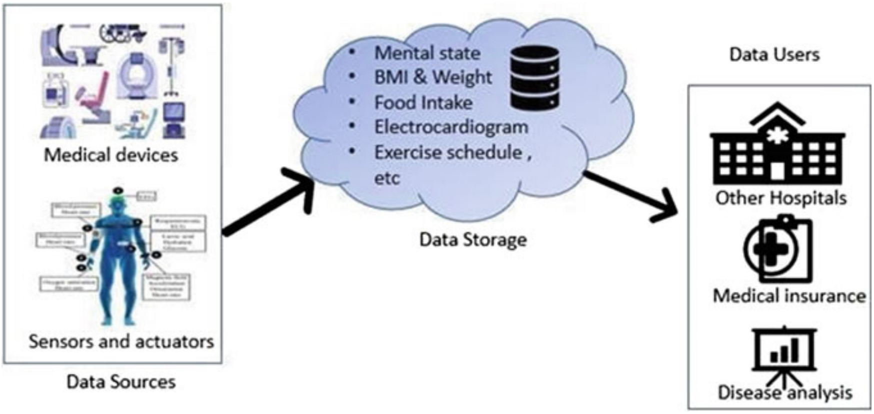
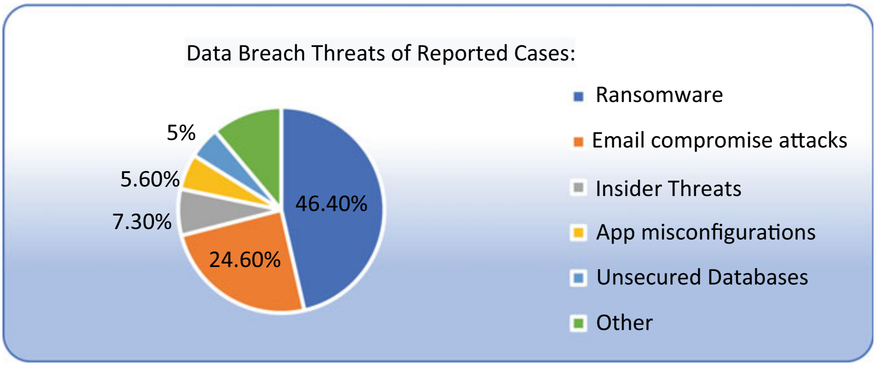


Fig. 1 Simplified architecture of MCPS



Reference: https://www.safetydetectives.com/blog/healthcare-cybersecurity-statistics/. This pie chart shows the data of various attacks that are performed on various medical organizations and the number of attacks that are increasing every year. This motivated us to try and come up with some better and improved mechanisms to store and share data

Thus, till now, we have seen and understood what exactly MCPS is and its various uses, the importance of MCPS. As we know the EHR is the vital component in these MCPS, there are various cyber-attacks done on MCPS to breach these data. The rate of attack is increasing day by day [22] as private patient information is worth a lot of money to attackers which are affecting the healthcare quality. Attackers are mainly of two types [18]:

Insider Threats: data breaches, ransomware attackers, social engineers.

External Threats: cyber-criminals, hacktivists, state-sponsored attackers. Information sharing, data integrity, and extraction are the major challenges that should be taken into consideration while designing MCPS. Personal health information is up to 50 times more valuable on the black market than financial information. The EHR contains [25] an electronic version of a patient's medical history which may contain progress notes, problems, laboratory data, medications, vital signs, past medical history, immunizations, etc. The data are of various formats. To store this data, an algorithm capable of encrypting all such formats must be designed along with this the algorithm must be cyber-attack resistant. It should provide an efficient way for the recognized authorities to access this data. MCPS is a decentralized system, and hence, security while securing data is a major concern. Medical records shared with patients; researchers improve the quality of the medical industry for secure and reliable communication and transaction of data; developing a network model is necessary [6].

2 Motivation

Because a compromise of the system can quickly damage the health and safety of patients, protecting MCPS from malicious assaults is critical. In the current period, MCPS is confronted with several key difficulties and obstacles, including many attacks, issues, and challenges that we must overcome in the current and next decade to provide efficient and reliable service to patients. MCPS requires smart health care and the attention of multiple research communities to address the concerns it has raised. Today's big story is CPS, which brings together a variety of professions to address safety, scheduling, efficiency, and, most critically, security [18]. The viability of CPS is dependent on ensuring secure data transfer over the network and proper machine functioning. Control systems, smart cars, smart grids, e-healthcare, and other applications use cyberinfrastructure based on Internet of Things devices. Cybercriminals, on the other hand, can hack these devices and take control of them from anywhere [10]. Terrorists, protestors, and organized criminal groups are all possible threats to the CPS and Internet-connected control systems. As a result, fresh attack detection [19] techniques, as well as new attack-resilient algorithms and architectures, are required in CPS. Attacks on such control systems must be protected and avoided at all costs.

3 Challenges

People who operate these systems are likely to have little awareness of system security and privacy, which is one of the security concerns that MCPS faces [2]. An attacker may use the existence of a patient's disease to blackmail the patient or conduct more serious attacks by intercepting the patient's medical device's communications using wireless hacking tools. A criminal hacker could use wireless equipment to inject false commands into a medical device, causing harm to a patient's health [27]. Medical equipment is vulnerable to a variety of attacks. Malware, security weaknesses, and threats are all on the rise. Over the years, there have been a substantial number of recalls [28]. The lack of security in continuous health monitoring system (CHMS) not only endangers the privacy of patients but also puts the user in danger [21] (i.e., the host).

Alkhushayni et al. [1] Data standardization and scope: Organizations must think about what information is saved in the blockchain and what information is not. The size of data stored on the blockchain is the greatest immediate challenge for the healthcare blockchain. A form submission of data to the blockchain, such as medical notes, could result in needlessly huge transaction sizes, compromising the network's performance.

While blockchain technology allows for speedier, near-real-time transactions, the cost of maintaining such a system is unknown. Traditional information systems and data exchanges need a significant amount of time and money to set up and run. This necessitates resources to diagnose difficulties regularly, update field parameters, and undertake backup and recovery procedures [9]. Since such systems are becoming increasingly interconnected and complicated, the main problem is to secure and develop the security, safety, and dependability of the MCPS devices. As the IoT phase progresses, one of the challenges is to deliver high-performance CPS systems

[18]. Concerning overcoming security and privacy problems, MCPS opens doors to numerous researchers, sectors, and so on. However, many cyber-physical systems have concerns such as security, privacy, and trust [20].

4 Existing Approaches of Securing MCPS

4.1 Based on Cryptographic Techniques

The author's Wang et al. [30] investigated fifteen devices from nine different manufacturers to demonstrate various real-life issues. They also discussed providing network security, e.g., medical devices could easily be attacked using MITM attack [23]. In these types of attacks, data being transferred are interrupted in between tempered and forwarded as if nothing has happened, and the receiver will never know if the data received are legit or not. A medical image standard, i.e., DICOM uses the application entity title (AE title) to identify the nodes that communicate during a transaction. Instead of focusing on all other nodes, it would be an easier task for the system to just focus on the two nodes communicating, and this is done using port number, AE title, and IP address. It suggested various protection mechanisms such as encrypted data storage and transmission, physical safeguards, system hardening, security guidance.

CPS is spanning the digital world with the actual world. The double dealing of such frameworks has broadly expanded in the new era. The authors Devi and Kalaichelvi [8] have discussed various aspects of CPS in the field of health care. It incorporates the actual parts known as biomedical sensors, data being stored on the cloud, cameras, and also persistently screens the progressions in the actual climate. The prerequisites of medical cyber-physical system to construct a proficient healthcare framework are security, privacy, data integrity, accessibility, interoperability, and data access control. Here, they examined different methods used to accomplish the protection and security of e-health records. E-health records keep up with the patient's clinical history and individual subtleties. Thus, it is important to ensure the data without spilling it to unapproved people. MCPS is a union innovation valuable to screen the clinical circumstance of patients at any place without restrictions. This paper investigated the different procedures used in medical cyber-physical systems, for example, remote body region organizations, cloud, electronic health records, big data, Internet of Things, and so on. The security necessities of MCPS are likewise examined.

According to the authors Kocabas et al. [15], this paper provides us with two things: 1. It surveys existing and emerging encryption scheme 2. It gives an evaluation of these schemes and later compares them and provides a result of the comparison. Every MCPS consists of 4 layers data acquisition layer, preprocessing layer, cloud storage, actuators layer.

Data are generated in the data acquisition layer through various medical biosensors and devices. Preprocessing layer acts as a middleware that connects medical biosensors to the cloud. It can also consists of an encryption algorithm that can be used to encrypt medical documents before storing them on the cloud [4]. Now, we have cloud storage which is a decentralized system and can be used to store and eventually share medical documents. Cloud can be used for processing, and also, we can use it for data analytics which could help facilitate decision support. In case of any emergency action layer or actuator, the layer sends signals to the emergency devices which could alert medical help as soon as possible.

The author's Zhang et al. [36] proposed a method that requires a third-party auditor to validate the data that are uploaded on the cloud by the owner or using a proxy. In this method, data owners can employ a third-party auditor periodically to validate the integrity of the documents being uploaded on the cloud. It designs a homomorphic MAC to compress medical data into small blocks to reduce size. It also designs elliptic curve digital signature algorithms along with symmetric encryption algorithm to ensure multiple layers to safety. The hash value of the document is generated before it is uploaded to the cloud and stored. Later, TPA verifies the hash value of the document after they are uploaded, and if these two hash values matches then we can be sure that the data has not been tampered, and we can rely on that data [33] (Table 1).

According to the authors Almohri et al. [2], the system is assumed to be a centralized design model. It consists of 2 subsystems one which directly interacts with practitioners and the second which deals with other components using human interaction. Generally, they are motivated to breach the privacy of patients by deleting or changing medical documents, tampering with the medical health records which could risk the life of the patient.

According to the authors Qiu et al. [26], this paper proposed a way in which we can securely store medical data even when both the cloud servers as well as keys are compromised. The method used in this paper is SEA, which is a selective encryption algorithm along with fragmentation and dispersion techniques. One big assumption has to be made that the physical device on the end user, i.e., his smartphone is trustworthy because the end user is given way too much power in his hand.

4.2 Based on Cloud Computing

According to the authors Verma et al. [29], this research studies the application of supervised learning algorithms in cyber-physical systems to recognize human actions using wearable biomedical sensors.

When compared to picture or video data, e-health sensors present a cost-effective answer for delivering sensor data. CPS paradigm proposes a method to identify activities using wearable sensors. Five wearable biomedical sensors are used for data collection (Table 2).

MCPS is a network system that connects various medical devices. The authors of the paper [13] have proposed architecture that consists of 5 phases:

Table 1 Comparative study on encryption techniques

S. No	References	Technique/algorithms used	Objective	Advantages	Limitations
1	Almohri et al. [2]	Cryptography, anomaly detection, system hardening	To secure MCPS against threats and attacks	Interaction with systems over an open network could be done by external component	Medical devices could be physically accessed to retrieve sensitive medical data
2	Tolgasoyata 2019 [15]	Attribute based encryption (Homomorphic encryption)	To perform secure/encrypted computation limits to share data	We don't have to observe the data before enabling the computation of meaningful operation	Practically impossible to use this since computational and storage requirements are pretty high
3	Qiu et al. [25]	Selective encryption algorithm	To secure medical data using SEA combined with fragmentation and dispersion	This method could also be used even when the cloud storage provider is not trustworthy	No algorithm is yet generated to determine which fragment is more vulnerable and needs to be secured first
4	Ara [3]	Cryptographic keys	To secure medical documents before storing them in a database	Data encryption is an enabler for achieving: flexibility compliance Information privacy	Difficult to keep a record of all the keys that are being used to encrypt the info which can negatively impact routine operations
5	Wang et al. [30]	Application entity title	To secure the network layer of the system for attacks like MITM, etc	To stop unauthorized access devices could use port no. Its IP address and AE title to identify the target node in the network	Address resolution protocol can still be used by attackers to create fake nodes with the same IP address and pretend to be original data owners

(continued)

Table 1 (continued)

S. No	References	Technique/algorithms used	Objective	Advantages	Limitations
6	Hasan et al.	Encryption algorithms or passwords	To secure the perception layer of the system which use RFID tags	It can track a large amount of data wirelessly without connecting to the reader's device	Can be used by anyone having access to it (physical displacement issue)

Table 2 Comparative study on cloud computing techniques

S. No	References	Technique/algorithms used	Objective	Advantages	Limitations
1	Lee et al. [16]	Cloud computing	To develop techniques and lay a foundation for effective, safe MCPS	Most of the system-Often combine only a few vital signs but this new system overcomes this problem	Occurrence of human error while taking the readings is possible
2	Verma et al. [29]	Cloud computing	Secure sharing data through biomedical sensors	1 Greater accuracy 2 Eliminates human errors 3 Patient is allowed to stay at home	-More power consumption -Cannot have a large size
3	Zhang et al. [35]	Data auditing using cloud	To share important medical documents	Allows original data owner to delegate the proxy	Expensive to maintain such a system
4	Xu [32]	Cloud-based MCPS	Novel proxy-oriented public auditing scheme to store data	Reduces communication and computational costs	1 Limited control 2 Multiple client

(1) Wearable Devices pre-deployment phase—Patient select WD, and hash function is generated for it

(2) Medical Devices registration phase—mark patient biometric and generating hash function for the MD

(3) Patient registration phase—verification of patient computation of pseudo-identity for user calculating hash function and sending a registration request to EHR
(4) Login and authentication phase—extract message, verify patient identity compute hash function encrypt the identity
(5) Addition phase for WD—patients authentication with cloud server.

The authors of this paper have discussed various flaws that existed in the previous auditing systems. Xu et al. [31] However, most of the schemes proposed so far work on public key infrastructure (PKI) cryptography in which managing certificates/documents is an issue. Few systems used bilinear pairing mechanism, but this method increases the computational cost which is why it was not a feasible method. Thus, to overcome all these problems, a new method is proposed which uses the ID-POPA scheme which prevents attackers from performing impersonating user attacks. This method has 5 participants, namely data owner, proxy, third-party auditor, medical cloud server, and private-key generator. The ID-POPA scheme can achieve a higher level of security with almost no additional computation costs. This scheme is a collection of 6 algorithms: setup, extract, proxy-key generator, data out, proof generator, proof verify.

4.3 Based on Blockchain

Process flow of data transaction was summarized by authors Alkhushayni et al. [1] as follows:

Data are encrypted first and then stored on the blockchain. The transaction is completed and is uniquely identified comprising of patient's public key, then to access this stored data, health organizations can use various API's and use data owner's public keys to retrieve encrypted data as and when required. Patients can share their private key with whoever they want and that key can be used to decrypt the stored documents. Thus, the data owners are placed at the center of the system, and they can grant or revoke access to anyone they feel like. This method uses the Ethereum framework instead of hyperledger.

According to the authors Chen and Tang [5], this paper is the revised edition of the previous one. In this paper, the security of the connected devices is improved using BAN logic. The traditional method provides security measures only when the devices are separated from each other, whereas BAN logic provides some additional features. The scheme can also be applied to various heterogeneous devices connected.

This approach proposed by the authors Mubarakali [17] is attribute-based encryption to provide security to the medical documents being stored. It collects patients' medical data at every moment and keeps a record of it [7]. It is then uploaded and stored on the cloud. This data are even shared with various medical organizations for consultation, and even insurance companies can access these documents for billing purposes. This method reduces 2.85 s in average delay, 1.69 s in system execution time, and it also improves the success rate by 28% compared to conventional techniques.

According to the authors of the paper [12], drawbacks of using blockchain are mentioned as follows: By storing a backup of all previous and current transactions at each node of the BN, a blockchain system provides data availability and transparency for all participants. This is almost unthinkable because it necessitates a large amount of data storage, which increases the number of blocks in BN. The whole history data are exposed to all participating AVs, which raises the question of data privacy. Future AV systems must consider this.

The authors of the paper [11] explain that to update and synchronize the data copies that are shared among multiple AVs; the blockchain consensus algorithm necessitates a significant amount of computer power and resources. However, AVs cannot execute these high-computing jobs, resulting in lower throughput and higher latency in the AV system (Table 3).

5 Challenges in Securing MCPS Data

5.1 Ensure the Safety and Confidentiality of the Patient Data

Patients' medical information is sensitive from a legal standpoint, which is why security is essential [18]. Administrative safeguards, physical safeguards, and technical safeguards are the three pillars of protecting protected health information. These three pillars are sometimes referred to as the three healthcare security safeguard themes. These themes include everything from computer location techniques to the use of firewall software to secure health information.

5.2 Secure Data Sharing

End-user data security and privacy in MCPS have been compromised as a result of recent cyber-assaults (MCPS). Traditional data encryption algorithms are built from the perspective of system architecture rather than from the perspective of end users. As a result of encryption algorithms moving data protection to key protection, data security and privacy could be a joke if we lose the key.

Table 3 Comparative studies on blockchain techniques

S. No.	Reference	Technique/algorithms used	Reference	Advantages	Limitations
1	Alkhushayni et al. [1]	Blockchain	To store and transfer EHR using blockchain	Network infrastructure security at all levels and transparency	Easy access to medical records could encourage insider privacy breaches
2	Cheng et al. [6]	Blockchain technology	To secure medical data using bilinear mapping	If this method is used central system won't have to be under a continuous workload.	If an entity owns 50% or more of the nodes, it can control the decision making in that network
3	Ali et al. [17]	Blockchain technology	Securing data using robust healthcare based BLOCKCHAIN (SRHB) approach	Minimize delays, Success rate is improved by 25% when compared with other techniques	It used Parity, Ethereum as its platform which is less productive than hyperledger and even has higher storage requirements
4	Chen et al. [5]	Blockchain	To provide a secure and reliable device identity authentication mechanism	Energy consumption of the device can also be saved service life is also increased	It only explains the existence of the medical alliance chain. Hyperledger fabric could still be used and we can keep on working

5.3 *Reduce Cyber-Attack*

As we know that cyber-attacks are increasing day by day, so to reduce attacks, improvement in MCPS is necessary. MCPS is designed to increase the effectiveness of patient care and give intelligent data to the caregiver and protect the safety of the patient. MCPS is being more widely utilized in medical cents to offer quality treatment, and they could be considered as potential platforms for monitoring and regulating a variety of elements of a patient's medical data and health.

6 Conclusion and Future Work

In this paper, we have reviewed various papers which used different techniques such as cryptography, blockchain, cloud computing, and so on. A summary of each paper has also been provided. We can see how each system has evolved over the years, and the comparison of those in terms of advantages, objectives, limitations can be referred from the analysis table. From the papers referred, we can say that cloud computing and blockchain ensure the basic security triad that is confidentiality, integrity, and availability which is not seen in cryptographic methods. Taking into consideration the above review, we will try to come up with a new technique in the future.

References

1. Alkhushayni S, Al-Zaleq D, Kengne N (2019) Blockchain technology applied to electronic health records. In: Proceedings of 32nd international conference
2. Almohri H, Cheng L, Yao D, Alemzadeh H (2017) On threat modelling and mitigation of medical cyber-physical systems. In: 2017 IEEE/ACM international conference on connected health: application, system & engineering technology, pp 114–119. IEEE
3. Ara A (2019) Privacy preservation in cloud based cyber physical systems. J Comput Theor Nanosci 16(10):4320–4327
4. Bhole D, Mote A, Patil R (2016) A new security protocol using hybrid cryptography algorithms. Int J Comput Sci Eng 4(2):18–22
5. Chen F, Tang Y, Cheng X, Xie D, Wang T, Zhao C (2021) Blockchain-based efficient device authentication protocol for medical cyber-physical systems
6. Cheng X, Chen F, Xie D, Sun H, Huang C (2020) Design of a secure medical data sharing scheme based on blockchain. J Med Syst 44(2):1–11
7. Deore S, Bachche R, Bichave A, Patil R (2021) Review on applications of blockchain for electronic health records systems. In: International conference on image processing and capsule networks, pp 609–616. Springer, Cham
8. Devi PV, Kalaichelvi V (2017) Security issues in medical cyber physical systems (MCPS)—a survey. Int J Pure Math 117(20):319–324
9. Dey N, Ashour AS, Shi F, Fong SJ, Tavares JMR (2018) Medical cyber-physical systems: a survey. J Med Syst 42(4):1–13
10. Gaikwad SR, Patil RY, Borse DG (2019) Advanced security in 2LQR code generation and document authentication. In: 2019 international conference on nascent technologies in engineering (ICNTE), pp 1–4. IEEE
11. Gupta R, Tanwar S, Kumar N, Tyagi S (2020) Blockchain-based security attack resilience schemes for autonomous vehicles in industry 4.0: a systematic review. Comput Electr Eng 86:106717
12. Hathaliya J, Sharma P, Tanwar S, Gupta R (2019) Blockchain-based remote patient monitoring in healthcare 4.0. In: 2019 IEEE 9th international conference on advanced computing (IACC), pp 87–91. IEEE
13. Hathaliya JJ, Tanwar S, Tyagi S, Kumar N (2019) Securing electronics healthcare records in healthcare 4.0: a biometric-based approach. Comput Electr Eng 76:398–410
14. Inbarani WS, Shenbagamoorthy G, Paul CKC (2013) Proxy re-encryption schemes for data storage security in cloud—a survey. Int J Eng Res Technol 2(1)
15. Kocabas O, Soyata T, Aktas MK (2016) Emerging security mechanisms for medical cyber physical systems, pp 401–416

16. Lee I, Sokolsky O, Chen S, Hatcliff J, Jee E, Kim B, King A, Mullen-Fortino M, Park S, Roederer A, Venkatasubramanian KK (2011) Challenges and research directions in medical cyber–physical systems. Proc IEEE 100(1):75–90
17. Mubarakali A (2020) Healthcare services monitoring in cloud using secure and robust healthcare-based BLOCKCHAIN (SRHB) approach. 25(4):1330–1337
18. Nair MM, Tyagi AK, Goyal R (2019) Medical cyber physical systems and its issues. Procedia Comput Sci 165:647–655
19. Patil RY, Devane SR (2017) Unmasking of source identity, a step beyond in cyber forensic. In Proceedings of the 10th international conference on security of information and networks, pp 157–164
20. Patil RY, Devane SR (2020) Hash tree-based device fingerprinting technique for network forensic investigation. In: AECT, pp 201–209. Springer, Singapore
21. Patil N, Patil R (2018) Achieving flatness: with video captcha, location tracking, selecting the honeywords. In: 2018 ICSCET, pp 1–6. IEEE
22. Patil RY, Ragha L (2011) A rate limiting mechanism for defending against flooding based distributed denial of service attack. In: 2011 world congress on information and communication technologies, pp 182–186. IEEE
23. Patil RY, Ranjanikar M (2021) Biometric authentication based smart bank locker security system. In: International conference on image processing, pp 298–308
24. Perdpunya T, Sukpradit A, Limpaporn O, Maliyaem M (2021) CPS based automation model prediction for inspection. In: 2021 13th international conference on knowledge and smart technology (KST), pp 108–112. IEEE
25. Qiu H, Qiu M, Liu M (2020) Secure health data sharing for medical cyber-physical systems for healthcare 4.0. IEEE Journal biomed Health Inf 24(9):2499–2505
26. Qiu H, Qiu M, Liu M (2020) Secure health data sharing for medical cyber-physical systems for healthcare 4.0. IEEE J Biomed Health Info 24(9):2499–2505
27. Rao A, Carreón N, Lysecky R, Rozenblit J (2017) Probabilistic threat detection for risk management in cyber-physical medical systems. IEEE Softw 35(1):38–43
28. Venkatasubramanian KK, Banerjee A, Gupta SK, Walls RJ (2017) A cyber-physical approach to trustworthy operation of health monitoring systems, pp 16
29. Verma H, Paul D, Bathula SR, Sinha S, Kumar S (2018) Human activity recognition with wearable biomedical sensors in cyber physical systems. In: 2018 15th IEEE India council international conference (INDICON), pp 1–6. IEEE
30. Wang Z, Ma P, Zou X, Zhang J, Yang T (2020) Security of medical cyber physical systems: An empirical study on imaging devices. In: IEEE INFOCOM 2020 IEEE conference on computer communications workshops, pp 997–1002. IEEE
31. Xu Z, He D, Wang H, Vijayakumar P, Choo KKR (2020) A novel proxyoriented public auditing scheme for cloud-based medical cyber physical systems 51:102453
32. Xu Z, He D, Wang H, Vijayakumar P, Choo KKR (2020) A novel proxyoriented public auditing scheme for cloud-based medical cyber physical systems. J Inf Secur Appl 51:102453
33. Yogesh PR (2020) Formal verification of secure evidence collection protocol using BAN logic and AVISPA. Procedia Comput Sci 167:1334–1344
34. Zhang Y, Qiu M, Tsai CW, Hassan MM, Alamri A (2015) Health-CPS: healthcare cyber-physical system assisted by cloud and big data. IEEE Syst J 11(1):8895
35. Zhang X, Zhao J, Mu L, Tang Y, Xu C (2019) Identity-based proxy-oriented outsourcing with public auditing in cloud-based MCPS 56:18-28
36. Zhang X, Zhao J, Mu L, Tang Y, Xu C (2019) Identity-based proxy-oriented outsourcing with public auditing in cloud-based medical cyber–physical systems, pp 18–28

Profiling Cyber Crimes from News Portals Using Web Scraping

Joel Christian, Sharada Valiveti, and Swati Jain

Abstract In the past few years and especially during the pandemic period everything is going online. Everyone is connected through the Internet. With this rapid surge, cyber crimes are also skyrocketing. The government has made efforts to cope with situation but that is not enough. India faced a loss of more than Rs. 1.25 trillion ($16 billion USD) due to cyber crimes. Online news are a reliable source of information which are always up to date and freely available. In this paper, we conduct a survey of web scraping related previous works. Web scraping is a method to gather information from websites. With this knowledge we have proposed a system of web scraping where we gather cyber crime related news articles. From this data we can classify the crimes into their respective categories according to their regions and time period. This can help law enforcement and also create awareness amongst common people.

Keywords Web scraping · Cyber crime · Online news · Information gathering · Profiling

1 Introduction

In this era of digitization where being online is a bliss, the threat of cyber crime is also increasing day by day. According to National Crime Records Bureau, India recorded a whopping 50,035 cyber crimes during the year of 2020. This increase in cyber crimes is 11.8% more than the previous year.

J. Christian (✉) · S. Valiveti · S. Jain
Department of Computer Science and Engineering, Institute of Technology, Nirma University, Chharodi, Ahmedabad 382481, Gujarat, India
e-mail: 20mcei03@nirmauni.ac.in

S. Valiveti
e-mail: sharada.valiveti@nirmauni.ac.in

S. Jain
e-mail: swati.jain@nirmauni.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_72

Cyber crime is a criminal activity that involves a computer and a network to exploit an individual, community, society, and the nation in general [24]. The cyber criminals are tech-savy people leading to sophisticated crimes. Criminals also evade being detected due to sophisticated attacks [45]. The most common cyber crime is phishing. Fake news, stalking and bullying of women and children, luring people using fake profiles, mobile banking fraud, identity theft, ransomware, publication/transmission of pornographic content, credit/debit card fraud, defamation, online gambling, etc. are some of the other activities leading to online crimes. Personal revenge, anger, fraud, extortion, causing disrepute, playing pranks, sexual exploitation, political motives and terrorism are some of the main reasons for performing cyber crimes [39].

Web scraping is the process of extracting large amounts of unstructured data from websites through software agents. Most of this data is in HTML format and is usually represented in a tabular format [41]. A web scraper can be divided into two parts, namely, crawler and scraper. A crawler follows the links provided and a scraper will extract the actual data from a given web page.

Real-time information gathering from portals and websites help the governing bodies to manage activities and set up fresh policies accordingly [11].

There are two types of profiling. First is person profiling which deals with the behavioural analysis of a particular person [40]. Second, data profiling is a process of reviewing data collected from a raw source, understanding the structure of that data and categorizing or summarizing the data in a readable format [32]. Data profiling is necessary because once analysed we get data which is free of anomalies.

Web scraping, can be used to gather information from available websites. This work aims to offer results in real-time as the news portals keep updating the contents on regular basis these days. This knowledge can also be used by the cyber specialists to predict any a pattern and present a counter-action to mitigate with cyber crimes in future.

2 Literature Review

This section explores the details of literature in the domain of Web Scraping. Lan and Soan [33] analysed the social media data to identify the impact of Covid-19 pandemic on the Vietnamese economy. Robihul et al. [30] created a risk diagnosis and mitigation system for Covid-19 in Indonesia. Thota and Ramez [44] used web scraping to get news stories about the pandemic and performed sentiment and emotion analysis.

Web scraping has many applications. But sometimes the traditional method can be hard to implement. There are some people who have worked to create a survey of different techniques and even created new techniques to gather information. In [34] the author proposes an intelligent and adaptive method of web scraping using Convolutional and LSTM deep learning networks. In [23] the researchers compares three different techniques to collect information relevant to social science. Computer vision and machine learning are growing rapidly. Dallmeier [10] Here the author proposes a computer vision-based method of web scraping instead of the traditional text-based method. In [7] the author gives two different approaches namely Summarization and Named Entity Recognition which use machine learning and natural language processing. In [31] proposes a method of web design scraping, which helps to understand the part of website with which a human is interacting. The papers [11, 43] give a survey of different techniques of web scraping, python libraries and its areas of application. In [15] the researcher give a demonstration of a language named OXPath which is designed specifically for web data extraction.

In social media Twitter data has been essential in diverse fields, so [17] proposes an enhanced method to collect twitter data. Sometimes the there can also be false information on social media, so it is essential to check the [12] credibility of that data. Social medias are also used by people for endorsement of their products [1]. Many a times it can be illegal to scrape some data from social media, so [26] describes the ethics of Facebook data collection.

Customer satisfaction is very essential in retail industry [19]. The shop owners can scrape customer reviews to improvise their products. A user can get the optimal prices by scraping the data from multiple E-commerce websites [5, 8, 37]. Stock market prices can also be collected to use it for price prediction using machine learning techniques [28].

Web scraping can be used for security purposes too [3, 46]. It can be used to collect cyber crime related data for research purposes. A web scraper implemented on the dark web would be really good [36]. In this paper they have implemented a system which can detect if a website is being accessed by a human or a bot. In [35] they have created and tested an intrusion detection system for website. Web crawlers have also been misused for malicious purposes. So to prevent this [47] created PathMarker, an anti-crawler mechanism which can detect is a bot is accessing your site.

The online news sites have plenty of data which is free to use and can be applied in various fields. For example, [29] used by law enforcement to identify drug trend markets, [22] natural language processing can be used to get summary of the whole article and [14] disaster management. In [6] a depth first search algorithm was created to analyse education related news.

The proposed work aims at identifying cyber crimes from the web portals and hence is an amalgamation of Intelligent and adaptive web data extraction system [34], news summarization [22, 27], Cyber crime research data collection [46] and applying required analytic for precise information compilation (Table 1).

3 Proposed Approach

In the proposed system we categorize cyber crimes to create awareness and to help law enforcement. The information regarding the crimes will be gathered from leading English news portals in India. Then the scraped data will have to be filtered to remove all the unnecessary and inconsistent data. And at the end, the user will be able see the types of cyber crimes happening according to the time period and location.

Table 1 Summary of literature review

S. No.	Application	Performance metrics	Dataset
1	Identify impact of Covid-19 [30, 33, 44]	Equity market comparison [33]; emotion versus count; comparison between manual and automated, python and R libraries [44]	Social media data [33]; statements and stories from news sites [44]
2	Intelligent and adaptive web data extraction system [34]; detecting a web scrapper on e-commerce site [37]	LSTM algorithm and convolutional network [34]; security mechanism using time and byte entropy analysis [5, 19, 37]	Product details amazon [34]; product prices on e-commerce sites [37]
3	To enhance Twitter data collection [17]	API versus Web extraction [17]; API versus normal web scraping; text, account age, verified, followings, followers [12]	Twitter data [12, 17]
4	To do data entry by text recognition [38, 43]	Comparison of precision, speed and accuracy of 3 libraries [38];	Images or text from 50 websites [38]
5	Detecting drug market trends; to help law enforcement and policymakers [29]	Location, pill logo, shape, colour, listing date [29]	User data from https://www.pillreports.net[29]
6	Web scraping using optical character recognition [10]	Comparison of text-based and computer vision-based web scraping [10]	Internet forums [10]
7	An algorithm for citation [42]; helps in citation [13, 15]	Comparison with two different algorithms [42]; python libraries: BS4 versus selenium [13]	Google scholar, web of science, scopus [13, 42]
8	Summarize news [22, 27]	Word frequency and weighted frequency [22]	Get news stories from Kazakh news site [22]
9	Website customization without programming knowledge [20, 31]	Create, extend and repair adapters by testing multiple websites [20]	Websites [20]
10	New technique for web scraping [4, 7, 11]	Summarization versus named entity recognition [7]	Diseases/pandemic related data [7]
11	Phishing URL detection [3]	Redirect links, form fields and download content [3]	URL crawling [3]
12	Evolution of OSINT data in research and education sector [6, 16, 18]	Research areas, regions, years [18]	Google scholar and NewsBank [18]

(continued)

Table 1 (continued)

S. No.	Application	Performance metrics	Dataset
13	Disaster management [14]	Term frequency-inverse document frequency vector	Online news articles
14	To answer complex questions related to computer science [25]	Jobs, salaries, degrees required	https://www.indeed.com
15	To help in learning and research materials [2]	API versus web scraping	GitHub
16	Cyber crime research data collection [46]	Site access, navigation, page loading and data gathering	Tor network and onion sites
17	Crop recommendation and optimal pricing for seeds and fertilizer [8]	Rainfall, temperature, location	E-commerce sites
18	Recipe finder [9]	Based of user defined ingredients	https://www.foodnetwork.com
19	Travelling recommendation system [21]	Content-based filtering, collaborative filtering, hybrid filtering, context-based filtering	Multiple travel and tourism websites
20	Ethical and legal issues; a scraping routine for public facebook posts [26]	Legalities in web scraping	Facebook data
21	Detect bots in web app [36]	Analysis of timing, movement, pressure and error patterns	Institutional web application log
22	Web intrusion detection system [35]	Error codes, data patterns, response timings, acknowledgements, no. of requests	Multiple websites running on different type of servers
23	PathMarker: an anti-crawler system [47]	Testing multiple crawlers by depth-first and breadth-first methods	Detecting google bots
24	Stock market price prediction [28]	Recursion method	National Stock Exchange India
25	Product endorsement on instagram [1]	Simple additive weighting	Instagram user data

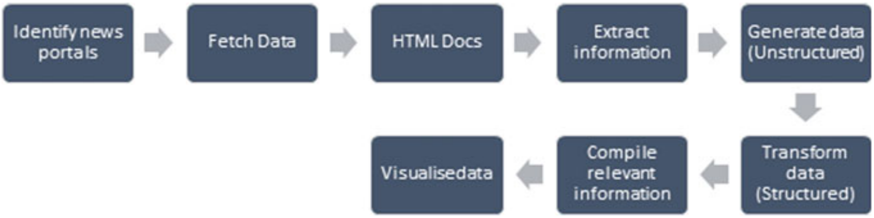

Identify news portals
Fetch Data
HTML Docs
Extract information
Generate data (Unstructured)
Transform data (Structured)
Compile relevant information
Visualisedata

Fig. 1 Web scraping process [31]

Python libraries like scrapy and BeautifulSoup4 will be used to scrape the data from news portals. Using these frameworks will ensure that we are not gathering any sensitive data as they follow the robots .txt security configuration. Also, they are much more efficient and faster than other traditional methods. The URLs of news articles related to cyber crimes will be gathered first. Then the headline and the article text will be scraped and stored. This stored data will then be accessible through a GUI web app. Through this web app the user will be able to visually see the types of crimes according to their respective cities, states and even particular dates. Thus this system will give us reliable information as soon as it is posted on the news portals (Fig. 1).

4 Implementation and Results

The web scraping model is implemented in Python using the Scrapy framework. There are two main methods to scrape website data, first is using the class and IDs used for CSS and the second is using the Xpath of the website. Here we have used the CSS selectors to get the details of each news article. The title, date, link, and summary of the article. Algorithm 1 shows the algorithm of the web crawler (Spider) that is implemented in the model. In Table 2 a sample result of the model is given. In this implementation we have scraped the website of NDTV news to get the list for cyber crime related articles. We have scraped the titles, dates and the links of the cyber crime related articles (Table 3).

Algorithm 1 Algorithm of Spider for web scraping

```
Require: $start\_urls[] \geq 0$
  while $i < length(start\_urls)$ do
      $response.css \leftarrow request(start\_urls[i]);$
      $i{+}{+};$
      while $articles \leftarrow responsse.css$ do
        $title \leftarrow article.css('title');$
        $date \leftarrow article.css('date');$
        $link \leftarrow article.css('link');$
```

Table 2 Results of web scraping the NDTV news portal

S. No.	Title	Date
1	Man blackmails MLA with fake objectionable video, arrested from Rajasthan	Tuesday November 23, 2021
2	Amid growing cyber crimes, Delhi to get cyber police station in every district from December	Tuesday November 23, 2021
3	400% rise in cyber crime cases committed against children in 2020: data	Sunday November 14, 2021
4	Beware of 'lucky draw' frauds ahead of diwali: Mumbai Police	Monday November 1, 2021
5	Russian extradited to us to face cyber crime charges	Friday October 29, 2021
6	Delhi police busts online investment fraud, arrests one accused	Sunday October 17, 2021
7	Over Rs. 12 crore of cyber fraud victims saved since 2018: centre	Thursday September 9, 2021
8	Goa police bust fake call centre, arrest 13 for duping US citizens	Wednesday September 8, 2021
9	NeGD signs pact with national law university Delhi, NLIU Bhopal to set up cyber lab	Friday September 3, 2021
10	Amazon, Google, Microsoft join US cyber team to fight ransomware	Friday August 6, 2021
11	Over 93,000 cyber crime cases reported from 2017 to 2019: centre to parliament	Tuesday August 3, 2021
12	Pegasus, other cyber product exports are for lawful use only: Israel defence ministry	Tuesday July 20, 2021
13	US sets up $10 million reward in fight against ransomware attacks; payout option in cryptocurrency	Friday July 16, 2021
14	4 arrested as Madhya Pradesh police bust crypto racket with Pakistan link	Tuesday July 6, 2021
15	Russian security chief says Moscow will cooperate with US against hackers: report	Wednesday June 23, 2021
16	Man blackmails MLA with fake objectionable video, arrested from Rajasthan	Tuesday November 23, 2021
17	Amid growing cyber crimes, Delhi to get cyber police station in every district from December	Tuesday November 23, 2021
18	400% rise in cyber crime cases committed against children in 2020: data	Sunday November 14, 2021
19	Beware of 'lucky draw' frauds ahead of diwali: Mumbai police	Monday November 1, 2021
20	Russian extradited to US to face cyber crime charges	Friday October 29, 2021

Table 3 Evaluation of the scraping time

S. No.	News portals	Items scraped (per page)	Time taken (s)
1	NDTV	30	0.465608
2	Times of India	20	0.692977
3	India Today	15	0.343795

5 Future Work

Although this type work has not been done in an automated manner, there are still many things that can be improved. The news portals are a reliable source but sometimes they cannot cover everything. This is where social media like Twitter comes into play. There are many accounts that post about cyber crimes. So, in addition to news portals we can gather information social media sites. Furthermore, we can use Natural Language Processing (NLP) to summarize the news articles so that we do not have to read the long articles.

6 Conclusions

This work focuses on classifying cyber crimes from the available news portals. The work can be extended to consider gathering information about many other crimes which go unnoticed due to several social reasons.

Through a system like this we propose to create awareness. So that people can understand the types of crimes that can happen to them. And eventually prevent these crimes from happening. It will also help the law enforcement to understand the criminal mindset and prepare themselves accordingly.

References

1. Akrianto MI, Hartanto AD, Priadana A (2019) The best parameters to select instagram account for endorsement using web scraping. In: 2019 4th international conference on information technology, information systems and electrical engineering (ICITISEE). IEEE, pp 40–45
2. AlMarzouq M, AlZaidan A, AlDallal J (2020) Mining github for research and education: challenges and opportunities. Int J Web Inf Syst 451–473
3. Almeida R, Westphall C (2020) Heuristic phishing detection and url checking methodology based on scraping and web crawling. In: 2020 IEEE international conference on intelligence and security informatics (ISI). IEEE, pp 1–6
4. Alrashed T, Almahmoud J, Zhang AX, Karger DR (2020) Scrapir: making web data apis accessible to end users. In: Proceedings of the 2020 CHI conference on human factors in computing systems, pp 1–12

5. AlZu'bi S, Aqel D, Mughaid A, Jararweh Y (2019) A multi-levels geo-location based crawling method for social media platforms. In: 2019 sixth international conference on social networks analysis, management and security (SNAMS). IEEE, pp 494–498
6. Endah Ratna Arumi and Pristi Sukmasetya (2020) Exploiting web scraping for education news analysis using depth-first search algorithm. Jurnal Online Informatika 5(1):19–26
7. Bhardwaj B, Ahmed SI, Jaiharie J, Sorabh Dadhich R, Ganesan M (2021) Web scraping using summarization and named entity recognition (ner). In: 2021 7th international conference on advanced computing and communication systems (ICACCS), vol 1. IEEE, pp 261–265
8. Chaudhari A, Beldar M, Dichwalkar R, Dholay S (2020) Crop recommendation and its optimal pricing using shopbot. In: 2020 international conference on smart electronics and communication (ICOSEC), IEEE, pp 36–41
9. Chaudhari S, Aparna R, Tekkur VG, Pavan GL, Karki GR (2020) Ingredient/recipe algorithm using web mining and web scraping for smart chef. In: 2020 IEEE international conference on electronics, computing and communication technologies (CONECCT). IEEE, pp 1–4
10. Dallmeier EC (2021) Computer vision-based web scraping for internet forums. In: 2021 7th international conference on optimization and applications (ICOA). IEEE, pp 1–5
11. Diouf R, Sarr EN, Sall O, Birregah B, Bousso M, Mbaye SN (2019) Web scraping: state-of-the-art and areas of application. In: 2019 IEEE international conference on big data (Big Data). IEEE, pp 6040–6042
12. Dongo I, Cadinale Y, Aguilera A, Martínez F, Quintero Y, Barrios S (2020) Web scraping versus twitter api: a comparison for a credibility analysis. In: Proceedings of the 22nd International conference on information integration and web-based applications & services, pp 263–273
13. Tran CD, Nguyen LD, Bui TD (2021) An author-based citation summary toolbox for google scholar. In: 2021 6th international conference on intelligent information technology, pp 73–79
14. Gopal Lakshmi S, Rekha P, Divya P, Vinodini RM (2020) Machine learning based classification of online news data for disaster management. In: (2020) IEEE global humanitarian technology conference (GHTC). IEEE, pp 1–8
15. Grasso G, Furche T, Schallhart C (2013) Effective web scraping with oxpath. In: Proceedings of the 22nd international conference on world wide web, pp 23–26
16. Hassanien HE-D (2019) Web scraping scientific repositories for augmented relevant literature search using crisp-dm. Appl Syst Innov 2(4):37
17. Henry D (2021) Twiscraper: a collaborative project to enhance twitter data collection. In: Proceedings of the 14th ACM international conference on web search and data mining, pp 886–889
18. Herrera-Cubides JF, Gaona-García PA, Sánchez-Alonso S (2020) Open-source intelligence educational resources: a visual perspective analysis. Appl Sci 10(21):7617
19. Udokwu CJ, Darbanian F, Falatouri TN, Brandtner P (2020) Evaluating technique for capturing customer satisfaction data in retail supply chain. In: 2020 the 4th international conference on e-commerce, e-business and e-government, pp 89–95
20. Katongo K, Litt G, Jackson D (2021) Towards end-user web scraping for customization. In: Companion proceedings of the 5th international conference on the art, science, and engineering of programming, pp 49–59
21. Sasi A, Kumar K, Birla V, Deep A (2020) Design and development of travel and tourism recommendation system using web-scraped data positioned on artificial intelligence and machine learning. Int J Adv Trends Comput Sci Eng 5670–5679
22. Kynabay B, Aldabergen A, Zhamanov A (2021) Automatic summarizing the news from inform.kz by using natural language processing tools. In: 2021 IEEE international conference on smart information systems and technologies (SIST). IEEE, pp 1–4
23. Li F, Zhou Y, Cai T (2019) Trails of data: three cases for collecting web information for social science research. Social Sci Comput Rev 0894439319886019
24. Loganathan M, Kirubakaran E (2011) A study on cyber crimes and protection. IJCSI Int J Comput Sci Issues 8(1)
25. Stephanie L, Jia Z, Monique R (2020) Utilizing web scraping and natural language processing to better inform pedagogical practice. In: (2020) IEEE frontiers in education conference (FIE). IEEE, pp 1–9

26. Mancosu M, Vegetti F (2020) What you can scrape and what is right to scrape: a proposal for a tool to collect public facebook data. Social Media+ Society 6(3):2056305120940703
27. Usha Manjari K, Rousha S, Sumanth D, Sirisha Devi J (2020) Extractive text summarization from web pages using selenium and tf-idf algorithm. In: 2020 4th international conference on trends in electronics and informatics (ICOEI)(48184). IEEE, pp 648–652
28. Maurya BBP, Ray A, Upadhyay A, Gour B, Khan AU (2019) Recursive stock price prediction with machine learning and web scrapping for specified time period. In: 2019 sixteenth international conference on wireless and optical communication networks (WOCN). IEEE, pp 1–3
29. Maybir J, Chapman B (2021) Web scraping of ecstasy user reports as a novel tool for detecting drug market trends. Forensic Sci Int Digital Invest 37:301172
30. Robihul MM, Arif B, Saniyatul M, Khusnul K, Nurul F (2020) Risk diagnosis and mitigation system of covid-19 using expert system and web scraping. In: (2020) International electronics symposium (IES). IEEE, pp 577–583
31. Namoun A, Alshanqiti A, Chamudi E, Rahmon MA (2020) Web design scraping: Enabling factors, opportunities and research directions. In: 2020 12th international conference on information technology and electrical engineering (ICITEE). IEEE, pp 104–109
32. Naumann Felix (2014) Data profiling revisited. ACM SIGMOD Record 42(4):40–49
33. Lan N, Soan D (2022) Policy response to covid-19 pandemic and its impact on the vietnamese economy: an analysis of social media. In: Financial and banking paradigm, Springer, Shifting Economic, pp 47–61
34. Patnaik SK, Narendra Babu C, Bhave M (2021) Intelligent and adaptive web data extraction system using convolutional and long short-term memory deep learning networks. Big Data Mining Anal 4(4):279–297
35. Ponmaniraj S, Kumar T, Goel AK (2020) Web intrusion detection system through crawler's event analysis. Int J 9(3):2503–2507
36. Rahman RU, Tomar DS (2020) A new web forensic framework for bot crime investigation. Forensic Sci Int Digital Invest 33:300943
37. Rahman RU, Tomar DS (2021) Threats of price scraping on e-commerce websites: attack model and its detection using neural network. J Comput Virol Hack Tech 17(1):75–89
38. Roopesh N, Akarsh MS, Narendra Babu C (2021) An optimal data entry method, using web scraping and text recognition, 2021 International Conference on Information Technology (ICIT). IEEE, pp 92–97
39. Hemraj S, Rao YS, Panda TS (2012) Cyber-crimes and their impacts: a review. Int J Eng Res Appl 2(2):202–209
40. Silvia S, Analía A (2009) Intelligent user profiling. In: Springer, artificial intelligence an international perspective, pp 193–216
41. Sirisuriya DS et al (2015) A comparative study on web scraping
42. Suganya E, Vijayarani S (2021) Firefly optimization algorithm based web scraping for web citation extraction. Wirel Personal Commun 118(2):1481–1505
43. Thivaharan S, Srivatsun G, Sarathambekai S (2020) A survey on python libraries used for social media content scraping. In: 2020 international conference on smart electronics and communication (ICOSEC). IEEE, pp 361–366
44. Thota P, Ramez E (2021)Web scraping of covid-19 news stories to create datasets for sentiment and emotion analysis. In: The 14th pervasive technologies related to assistive environments conference, pp 306–314
45. Tounsi W], Rais H (2018) A survey on technical threat intelligence in the age of sophisticated cyber attacks. Comput Sec 72:212–233
46. Kieron T, Sergio P, Collier B (2020) A tight scrape: methodological approaches to cybercrime research data collection in adversarial environments. In: (2020) IEEE European symposium on security and privacy workshops (EuroS & PW). IEEE, pp 428–437
47. Wan S, Li Yue, Sun K (2019) Pathmarker: protecting web contents against inside crawlers. Cybersecurity 2(1):1–17

Privacy Preserving Outsourced *k* Nearest Neighbors Classification: Comprehensive Study

Vijayendra Sanjay Gaikwad, K. H. Walse, and V. M. Thakare

Abstract Cloud computing is now an integral infrastructure for individuals and companies to store data and provide services that leverage this data. Thus, it is a routine for data owners to outsource their data and data operations to a public cloud that serves certain mining services. In order to preserve privacy, the valuable outsourced data has to be encrypted before outsourcing to the public cloud. Because of this, it is now required that classification is performed using the encrypted sensitive data in an un-trusted outsourced environment, like cloud, and at the same time maintain its privacy. Since the *k* nearest neighbor (*k*NN) classification techniques are considered to be a basic module for many common data mining tasks, establishing an efficient and privacy preserving *k*NN approach will form a concrete platform for other fundamental data analysis operations. Due to its significance in different mining applications, many solutions have been proposed to solve the outsourced *k*NN classification over encrypted data problem. In this paper, we present a comprehensive study of such solutions. As per our study, most of these solutions either struggle to completely achieve the privacy requirements of outsourced classification or they induce heavy computational cost which makes them impractical for use in real-world applications. In this context, we compare the existing state-of-the-art solutions in terms of performance with standard and scaled datasets. It is observed that most of the efficient solutions for the outsourced *k*NN classification problem collapse as the dataset is scaled both attribute-wise and size-wise and their performance reduces significantly. We have attempted to identify such significant issues in this research area and address them through our future subsequent research work.

V. S. Gaikwad (✉) · V. M. Thakare
P.G. Department of Computer Science & Engineering, Sant Gadge Baba Amravati University, Amravati, Maharashtra, India
e-mail: vij711@gmail.com

V. S. Gaikwad
Department of Computer Engineering, Pune Institute of Computer Technology, Pune, Maharashtra, India

K. H. Walse
Department of Computer Science & Engineering, Anuradha Engineering College, Chikhli, Maharashtra, India

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_73

Keywords Encrypted database · Privacy preservation · Outsourced data · k nearest neighbor classification · Homomorphic encryption

1 Introduction

Organizations are continuously getting attracted toward leveraging the cloud potentials for analyzing their data. However, the only concern is, "How will the privacy of their data be preserved while performing the data analysis?" Certainly, much advancement has been made recently in this regard. But most of the recently proposed privacy preserving solutions are really struggling to establish the right balance between the privacy levels availed by these solutions and their computational costs. In this paper, we are showcasing some of our in-depth findings made through extensive literature review of the various privacy preserving solutions that were recently proposed to solve the outsourced classification problem. The motivation for this literature review is to identify the research gaps, issues and challenges in this research area of privacy preserved classification in outsourced environment, which we will attempt to address through our research work.

Data stored in the outsourced environment like a public cloud is encrypted. So, security-wise the arrangements made by the cloud service providers are so far good. But, when it comes to preserving the privacy, there are still some questions in the minds of users. This is because it may be required to perform data mining over the user's encrypted data in the cloud environment. Such a mining at some stage requires decryption of the encrypted data, and this is when the data gets revealed to the cloud.

In our earlier study [1], we mentioned that the common means available for privacy preserving data mining (PPDM) are attribute randomization, anonymization technique and cryptographic technique. As stated in [1], randomization method is efficient, scalable and a bit more accurate and provides greater privacy. The performance of PPDM techniques on various parameters is as specified in our previous study [1] is mentioned in Table 1.

In most of the recent solutions to the outsourced *k* nearest neighbor classification problem [2–5], the cryptographic technique is combined with other techniques so

Table 1 PPDM techniques evaluation

Criteria→	Computing cost	Accuracy	Privacy preservation	Scalability
Technique				
Randomization method	Least	Highest	Highest	Highest
Cryptography method	Highest	Highest	Highest	Least
Anonymization method	Least	Average	Average	Least

as to achieve a privacy preserving scheme. Moreover, majority of the solutions [6–11] are built over the two-cloud honest-but-curious setup, described in [2, 3] to avoid disclosing the data access patterns. All the schemes use partial homomorphic encryption schemes like Paillier cryptosystem [12] or ElGamal cryptosystem [13]. This review considers the *k* nearest neighbor (*kNN*) classification techniques as a significant data analysis method. This is because the *kNN* classification is considered to be a basic module for many common data mining tasks, and hence, establishing an efficient and privacy preserving *kNN* approach will form a concrete platform for other fundamental data analysis operations. Due to its significance in different mining applications, most of the solutions reviewed in this work have been proposed for the *k* nearest neighbor classification problem using encrypted data.

2 Literature Review of Recent Solutions

If the classification queries are processed in the cloud environment, then the primary concern is to prevent the query from being revealed to the cloud. As mentioned in our preliminary study [1], the attributes values of the user query must be encrypted, and during the entire classification process, they must never be disclosed. Also we quoted in [1] that, “any number of operations are allowed to be performed using fully homomorphic cryptosystems (FHE) over encrypted data with comparatively less efforts” [1]. The scheme formulated by Gentry [14] is able to execute arbitrary type of functions on the encrypted data and that too any number of times, without ever decrypting it. But, FHE proves to be computationally costly. So, use of FHE in serving an online classification request is practically impossible.

We have also discussed the shortcomings of the S*kNN* protocol [2] and P*kNN* protocol [3] in our study [1]. While following the S*kNN* protocol [2], the squared Euclidean distances computed, *k* smallest distances and other intermediate results are disclosed to the un-trusted cloud. Also, the database tuples corresponding to the *k* nearest neighbors of the user’s classification query get disclosed to the cloud and user. Disclosure of database records used to generate the classification outcome results in invasion of privacy. In PP*kNN* protocol [3], the privacy of the *k* nearest neighbor database records is preserved from the cloud and user. So, PP*k*NN protocol determines the *k* nearest neighbors by preserving privacy through many other secure sub-protocols. Table 2 shows the privacy performance of the protocols in [2] and [3]. The requirements of a privacy preserving *k*NN classification as described in [3] are mentioned in Table 2 and whether or not these protocols preserve or disclose the data that is mentioned.

However, the evaluation mentioned in [3] shows that the maximum of the computational time required by PP*k*NN protocol, i.e., almost 67%, is incurred by a secure minimum protocol. So, through experimental analysis in [3], PP*k*NN is clearly computationally costly protocol and hence not very practical. So, reducing this computational cost, either by inducing parallelism or refining the protocol itself,

TABLE 2 Comparison of studied privacy preserving classification protocols

Criteria→	User query	Original database attribute values	Intermediate results	Records corresponding to the k-nearest neighbors of query	Classification result	Computational cost
Protocols						
S*k*NN [2]	PRESERVED	PRESERVED	DISCLOSED	DISCLOSED (to the Cloud)	COMPUTABLE	MEDIUM
PP*k*NN [3]	PRESERVED	PRESERVED	PRESERVED	PRESERVED	PRESERVED	HIGH

in order to propose a practical and feasible solution to privacy preserved data mining problem, is the current challenge.

Khodaparast [6] assumes that data can be distributed between two parties. Both parties collaboratively try to create a random decision tree classifier on complete dataset, but due to potential privacy issues, sharing of the original datasets may not happen. The algorithms which are proposed in [6] allow private collaboration between the two parties, in such a way that both the parties try to determine the classifier's structure based on complete data, without ever realizing the other party's information. In [6], authors have securely built a random decision tree classifier where the data to be utilized for classification is distributed among two parties. However, the solution proposed in [6] requires at least two parties for data storage. The major concern is that the horizontally or vertically distributed data is in plain form.

In [7], authors have proposed an outsourcing scheme for support vector machine (SVM) classification in public clouds which is efficient and also preserves privacy. This SVM classification in public cloud uses order-preserving encryption (OPE) [15, 16] for preserving confidentiality of the SVM classifier and the users' features vector (input) and prediction results (output).The upper and lower boundaries of hyper-rectangles extracted from a SVM classifier are encrypted using order-preserving encryption. Also, the user encrypts every dimension of the feature vector (that is to be classified) using order-preserving encryption. On the public cloud, this feature vector is then compared with each hyper-rectangle's upper and lower boundaries to determine the encrypted class label as result. However, SVM classification requires inequality comparisons to find the classification result. Order-preserving encryption (OPE) method allows performing inequality comparisons on the encrypted data. However, it is not able to perform more complex operations on the encrypted data. *kNN* classification requires more complex operations other than inequality comparisons. In case of OPE scheme used for *kNN* classification, it will require the intermediate data to be decrypted, which will leak these intermediate results and access patterns to the cloud.

In [5], a scheme for privacy preserved *kNN* classification in outsourced environment is proposed that covers all of the privacy requirements for an outsourced *kNN* classification that are specified in [3], but it is not a semantically secure solution and hence is prone to "known ciphertext attacks". However, their work related to outsourced privacy preserving *kNN* classification in [12] achieves semantic security. The PPKC protocol [5], however, does not provide the class label corresponding to the user's query. Instead, it only results in the class labels of all the *k* nearest tuples corresponding to user's query. The *k* nearest class labels are returned to the querying user. Thus, PPKC provides a solution to the secure *kNN* problem, but not to the privacy preserving *kNN* classification problem. It also reveals the *k* nearest class labels to the querying user, which does not satisfy the privacy requirement as stated in [3].

To overcome the efficiency issues of the state-of-the-art PP*kNN* protocol [3], another more efficient protocol [4] has been proposed that works in the same experimental setup as used in [3]. The state-of-the-art protocol PP*kNN* is very complicated and has high computational cost. The efficiency and performance issues of PP*kNN* that were incurred by the computationally costly sub-protocols have been overcome to a certain extent by the set of protocols proposed in [4]. It uses Paillier cryptosystem [12] to encrypt the user query and database tuples and ElGamal cryptosystem [13] to encrypt the classification labels corresponding to each database tuples. This allows the classification labels to be re-encrypted.

In [8], authors have proposed a new solution in which an outsourced Naive Bayesian classifier accepts encrypted query and preserves the confidentiality of the classifier, the instances of query and result. However, according to the latest study done in [17], it induces heavy computational costs (due to the less efficient Paillier cryptosystem), so the communication time between the users and the server is affected. Also, the bandwidth gets affected. This has a direct impact on users' experience and can be lagging. Moreover, as proven in [17], with the STC attacks users can acquire the knowledge of the classifier. Extensive work on the same semi-honest cloud setup for classification of kNN query securely has been done in [18]. In this approach, due to the use of somewhat homomorphic scheme, the encrypted squared Euclidean distances computed by one cloud server for each record are processed through a permutation function before they are sent to the other cloud server for sorting. The authors claim that the confidentiality of specific distance values, query results and access patterns can be protected by this approach. This sorted sequence of encrypted distances is sent back to the first cloud server without learning the actual values. However, as per the cryptanalysis presented by Murthy [19], even if it is impossible to find out the coefficients or the inputs by just using the outputs, they have shown that it is possible to get back the inputs (up to a constant difference) of up to 32 bits size when output is greater than polynomial degree.

In [9], authors have proposed a classification protocol that is efficient and secure and also works with encrypted data that uses vector homomorphic encryption for row-wise data encryption. Their analysis of protocol security indicates that the proposed protocol provides protects data and query record from being disclosed and hides data access patterns. Yang et al. proved that the protocol achieves the same 98% accuracy with as compared with *kNN* classification over plaintexts. However, when we set the precision to single digit, at that time the accuracy reduces significantly to almost 57%. Although this scheme is quite efficient, the insecurity of vector homomorphic encryption used in this scheme is recently proven in [20].

Also study in [10] claims that it is unrealistic to fully trust the query users. So, the private key (used for encryption and decryption of data) owned by the data owner need not be known to users. A protocol for securely computing the *k* nearest neighbors of the user query that utilizes multiple keys has been proposed in [10]. In this, the data owner and every querying user have separate key in a way that they are not required to share their key. Although this scheme preserves data and query attribute privacy and refrains the data owner from direct participation, still the access patterns are not protected.

The sub-protocols in [3] used collaboratively by the two clouds are redesigned in [11]. This is a data outsourcing schemes based on secret sharing. As per their experimental results, the scheme is efficient than [3], but is based on shared secrets, and as per the study of Oktay [21] and Dautrich [22], secret sharing techniques make an assumption that participating servers do not collude. Also, efficiency analysis shows that the dataset dimensions impact the performance of the secure minimum, secure maximum and secure frequency protocols proposed in [11]. Moreover, for the secure frequency protocol, the variety in class labels also affects the overall performance of these protocols.

Hsu has proposed another privacy preserving kNN scheme for searching over the encrypted dataset outsourced on cloud [23]. It allows the cloud to determine the *k* nearest data points from an encrypted dataset that are closer to the encrypted user query. Then the search results are returned to the querying user. Here the authors claim that the secret key always remains with the data owners and the secret key owned by data owners is not required to be shared with others. The authors have tried to overcome limitations of the earlier state-of-the-art work such as the need for distributing the secret key by the data owner to others, providing some storage at the user's end, using non-colluding cloud server setup. However, this approach lowers the level of security. So, the privacy of the outsourced dataset and query is protected, but it will disclose the accessing pattern of data to the cloud server. According to their study, they claim that if the data is encrypted, then the disclosure of the accessing pattern of data is not a concern, which is questionable.

Another recent scheme proposed in [24] uses an encrypted k-dimensional tree (kd-tree) proposed in [25] to optimize the traditional kNN algorithm. kd-tree data structure arranges the dataset's k-dimensional instances into a binary tree structure. When the nearest neighbor of a query record is to be found, this structure is then traversed. However, this scheme is not semantically secure. Also, the secret keys are communicated to the query user, which can affect the confidentiality of data owner's data. Moreover, as the dataset dimensions increase beyond 12 (especially at 18), the accuracy of this scheme significantly reduces. Thus, the use of kd-tree algorithm along with scaled attributes of the dataset is yet to be investigated.

In [26], a privacy preserving *kNN* classification protocol is called PkNC that utilizes the sub-protocols in [3], but in a modified way so as to support their parallel execution in order to find the encrypted classification result. In all of the privacy preserving *kNN* classification solutions, as a part of selecting the closest data points, if there are same elements in the sequence of Euclidean distances, then any one of the same elements is returned as result. However, the SkSE protocol considers multiple same elements as result. It is experimentally proven that the execution time of the PkNC protocol drops significantly regardless of the total number of closest neighbors, i.e., *k,* and the execution time increases gradually as compared to the linear increase of PP*k*NN [3] when the key size of Paillier cryptosystem advances. But, the actual downfall of the PkNC protocol is seen when the dataset size grows. As the number of data records increases, the execution time also increases linearly. This is mainly because there is practically a restriction on thread to be used for parallelism.

The privacy of the encrypted data records and users' query is assured in [27]. They have devised indexed searching method that works with encrypted data to perform data filtering that provides huge efficiency in query processing. This scheme searches through the computed encrypted Euclidean squared distances and does not disclose the data access patterns. The analysis mentioned in [27] proves that this protocol preserves data privacy, and the query processing is far better than the existing ones. This scheme uses the encrypted kd-tree search as an indexing structure since it is more favorable when indexing data with multi-dimensions. kd-tree search is more efficient as compared to the linear search through the encrypted distances. However, this scheme is not semantically secure. Also, as stated earlier, the use of the kd-tree algorithm along with scaled attributes of the dataset is yet to be investigated. Also, the impact of attribute scaling on the computation time and the classification result's accuracy is also a concern.

The work in [28] attempts to protect the confidentiality of the original data and user query privacy and does not disclose the legitimate user's location. The authors have utilized the Moore curve [28] in order to convert the spatial data into a vector. Also, the authors have utilized the AES algorithm for encrypting the original data. As per cryptographic transformation, their method reduces the overhead in communication and hence can provide the k closest neighbors of a query efficiently while protecting location information and spatial data. In order to obtain the accurate kNN points of interest, the authenticate user (AU), i.e., the trusted party, decrypts the encrypted transformed data index (TDI) table records to obtain the points of interest (POIs) and then computes the difference between the query point and all the POIs. This decryption at the trusted party or AU allows the AU to gain knowledge of difference values which can be used later to plot attacks. Moreover, this encryption used is AES which is not semantically secure and induces additional privacy issues. The authors have mentioned in their future scope about the need of improving the security of their proposed protocol by using homomorphic encryption.

The possibility of decision tree's training in efficient manner and evaluation for classification purpose in the cloud environment and accomplished privacy preservation has been investigated in [29]. The authors have asserted a theory that the cloud cannot extract out the most useful attributes from an encrypted dataset, and hence, they have proposed a way to train the decision tree without dataset splitting. The protocols designed by authors work under the same system model and consider the same threat model as used in [3]. Their experiments reveal that the computational and communication cost of the proposed protocols grows with the vector length (i.e., with the increase in number of attributes) and the number of classification labels. A very important fact noticed is that their protocols were tested using very small datasets which have 24, 100, 120 and 958 instances, respectively. It is very evident that training the model (without any bias) requires huge datasets, and hence, these protocols need to be investigated using standard machine learning datasets.

3 Issues with Recent Solutions

Currently, many web applications that regularly analyze the user generated do provide some basic privacy information about how and to what extent they access our data. This data may be collected to keep a track of our visits to that particular website. They utilize this information to deliver web pages (i.e., content) which is tailor made for its users. Also it provides the websites with information regarding your location. Many websites use third-party advertising companies to serve ads when we visit their websites. It is assured by the privacy policies of many web applications that they do not access the individual's information like name, address, email address or telephone number of their visitors, but this is highly questionable because we even get notifications on our email about the goods and services of our interest later. This proves that most of these web applications utilize the user's data directly (i.e., without any preprocessing for preserving privacy). So, a user has to agree to the data usage policy of the websites (may be forcefully) and allow their mining party to use this personal and sensitive data. Also, in day-to-day usage of gadgets, a lot of data is continuously uploaded to the cloud storage through our devices. This data is then utilized to serve us with many different services. This usage is nothing but the mining and analysis performed on the device generated data. Moreover, this data can be further utilized by other third-party cloud services providers for providing us with some other additional services, which are not intended by the predefined use of those devices. Obviously, this can only be done after the consent of user and if the intended cloud storage has collaborated with that third-party cloud services provider. Now this may create a problem as our sensitive data might be disclosed during this process.

Privacy preserved classification is a fast-growing research area, but there are certain issues which can be considered as directions for future research. The major issues in this research area are maintaining privacy of encrypted data, preserving privacy of a user's query attribute values and keeping data access patterns reserved from cloud, which still remain vital research issues. Despite so many solutions proposed till date, every solution misses out on either of these privacy requirements for classification over encrypted data. Other issues identified through the literature review are as follows.

3.1 Data Decryption Needed at Some Point of Time

Since FHE [14] is computational costly and not practically feasible to use, partial homomorphic encryption scheme is the only choice for encrypting the data. However, the problem with partial homomorphic encryption is that it allows limited operations over encrypted data; hence, data has to be decrypted at some stage. Hence, there is a possibility that either of privacy requirements is not conformed. This remains a major issue in the given research area.

3.2 Heavy Computational Overhead

Most recent and relevant schemes either cannot simultaneously achieve all the three privacy requirements or they induce heavy computation overheads due to computation performed on encrypted data. Hence, these schemes are not practical enough in real-world scenarios.

3.3 Unexplored Multi-Label Classification

The result of *kNN* classification, on certain occasions, can be more than one class labels. However, most of the recent and proven solutions in [3–5, 7, 9–11, 18, 24] and [29] are only able to output one class label, resulting in an error.

3.4 Reduced Efficiency with Large Datasets

It is a common phenomenon among all recent solutions that the computational and communication costs grow with the increase in number of instances, attributes and classification labels of the dataset. The kd-tree algorithm can be leveraged as an alternative to the traditional kNN algorithm. It arranges the dataset's k-dimensional instances into a binary tree structure [27]. When the nearest neighbor of a query record is to be found, this structure is then traversed.

However, it is proven in [24] that as the dataset dimensions increase beyond 12 (especially at 18), the accuracy of this scheme reduces significantly. This requires further research on the influence of the kd-tree algorithm and attributes of the dataset on the classification results and computational cost.

The privacy preserving *kNN* classification protocol called PkNC [26] utilizes the sub-protocols proposed in [3] and executes them in parallel to find the encrypted classification result. However, the actual downfall of the PkNC protocol is seen when the dataset size grows. The execution time grows linearly with the number of data instances, which is mainly due to the practical limit of thread for parallel computation. It is very evident that training the classifier requires huge datasets, and hence, the existing schemes need to be investigated using standard machine learning datasets.

3.5 Inability of Differential Privacy to Solve Complex Problems

Differential privacy [30] transforms the dataset by adding statistical noise on the original data to hide any direct information gain from the general population. Usually, it is used to perform simple computations like, count, min, max, etc. Privacy preserving kNN classification is a much complicated data processing problem, and adding noise at every stage will totally destroy the usefulness of data. Hence, differential privacy cannot be directly used for kNN classification that too in an outsourced environment.

3.6 Need for a Two-Cloud Setup

Partial homomorphic encryption cannot perform any arbitrary function on encrypted data. So, while performing classification over such encrypted data, decryption of data at some stage on the cloud is unavoidable so that decrypted data can be processed and based on which classification is done. But, this decryption of encrypted data causes the cloud to acquire knowledge about sensitive data. Such knowledge gained by cloud invades the privacy of user sensitive data. In a two-cloud setup, encrypted data is randomized on one cloud and sent to other cloud where it is decrypted. Thus, the two clouds collaboratively perform the steps involved in kNN classification while maintaining privacy of intermediate data.

4 Methodology for Privacy Preserving kNN Classification

It is challenging to perform classification of user's queries using the encrypted database records. Moreover, this task becomes even more difficult when the classification is to be done on an un-trusted cloud and that too without ever decrypting the data. When the encrypted database and classification of the process itself are outsourced to an un-trusted, third-party cloud, then there is a need for a collection of secure and privacy preserving sub-protocols that collaboratively execute such a classification job.

We consider the same scenario as described in our previous work [1], where the database consists of n records and m attributes. The mth attribute is considered to be the class label of any record, denoted as c. We consider t is a database record and E_{pk} $(t_{i,j})$ corresponds to an encrypted record value in such a way that $1 \leq i \leq n$ and $0 \leq j \leq m$. So, the data owner encrypts the database records attribute-wise using partial homomorphic scheme [12], and the encryption function E_{pk} is semantically secure [2]. The data owner then outsources the encrypted database, denoted by *EDB* along with the classification process and hereafter has no intervention in the outsourced classification process.

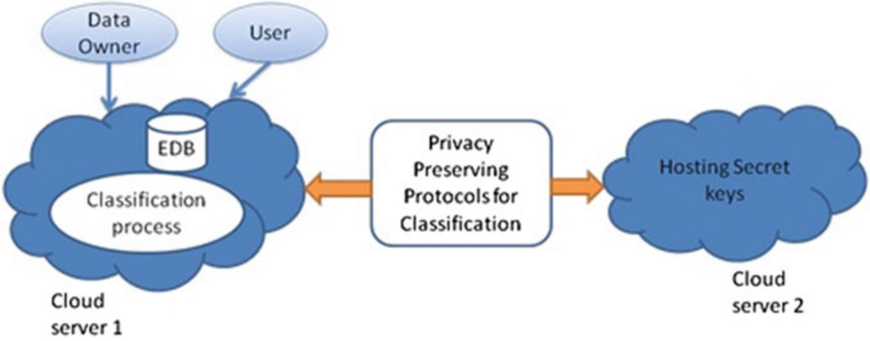


Fig. 1 Methodology for privacy preserving classification in two-party setup

Considering the above described scenario in Fig. 1, fetching the k nearest database records with respect to the user's classification query without disclosing query attributes and intermediate results to both the clouds is the main aim of the privacy preserving kNN classification. Figure 1 depicts the scenario in which the privacy preserving classification is performed. The user at first needs to prepare a query Q, comprising m attributes

$$Q = (q1 \ldots qm) \tag{1}$$

The user then encrypts the query using a partial homomorphic scheme [12] and sends it to cloud-1 ($C1$). After this, cloud-1 ($C1$) and cloud-2 ($C2$) collaboratively execute a set of secure and privacy preserving protocols in order to fetch the k database records that are nearest to the user's classification query Q. Encrypted query is denoted as follows:

$$E_{pk}(Q) = \big(E_{pk}(q1) \ldots,\ E_{pk}(qm)\big) \tag{2}$$

Upon receiving E_{pk} (Q) from user, cloud-1 ($C1$) with private inputs of E_{pk} (Q) and E_{pk} (ti) and cloud-2 ($C2$) with private key denoted as, sk, will first execute a secure protocol to compute the squared Euclidean distances between Q and each ti, denoted as di. This protocol results into a vector consisting encrypted distance values, denoted as, E_{pk} (di). $C1$ does not gain any information about these distance values since they always remain encrypted. Thereafter, every element of the vector is randomized using additive homomorphic property [12]. Then, the vector carrying the randomized encrypted distances paired with corresponding index value is sent by $C1$ in following format

$$\big\{\big(1,\ E_{pk}(d1 + r)\big),\ \ldots,\ \big(n,\ E_{pk}(dn + r)\big)\big)\big\} \tag{3}$$

to $C2$ where, for each pair (i, E_{pk} (di + r)), $1 \leq i \leq n$. $C2$ receives the vector {(1, E_{pk} ($d1$ + r), …, (n, E_{pk} (dn + r)} and decrypts it with the private key sk to get a sequence of randomized distance values, D_{sk} (E_{pk} (di + r)). $C2$ then sorts this sequence and gets the k smallest randomized distance values as

$$[(d1 + \mathrm{r}),\ \ldots,\ (\mathrm{dk} + r)] \tag{4}$$

$C2$ then forms a vector of indexes corresponding to these k randomized distances as [$i1$,…, ik] and sends this vector to $C1$. Although $C2$ performs computation on pain data, still $C2$ does not acquire any information about the intermediate result since data received by $C2$ is already randomized on $C1$. However, access pattern in the form of index values may get revealed.

Now $C1$ fetches the database tuples corresponding to the received index vector, [$i1$,…,ik]. However, due to these index values the corresponding database records may be identified by the cloud $C1$. So, although actual database records are not revealed to $C1$, it may learn which records in the database are similar to the user query. Furthermore, $C1$ then adds a unique randomization factor r_i to each of the k nearest database records individually using additive homomorphic property [12], and these randomized database records are then sent to $C2$. At the same instance, a vector comprising randomization factors [r_1……r_k] is sent by $C1$ to the querying user. Upon receiving the k randomized database records from $C1$, these records are decrypted by $C2$ and then sent to the querying user.

Once the querying user receives the randomized decrypted records from $C1$ and the vector comprising randomization factors [r_1……r_k] from $C2$, the user deducts the randomization factor r_i from each of the k plain database records attribute-wise. But again here, the k nearest database records may get revealed to the user. As a solution to this issue, we are currently working on a permutation function that will shuffle the encrypted elements of the vector so that access patterns are also preserved.

5 Comparison of Recent State-of-The-Art Solutions

For our comparative study, we have chosen a few most recent and relatively secure solutions to the outsourced *kNN* classification problem. Firstly, we evaluated these solutions based on the size of the dataset used for experimentation (i.e., number of instances and attributes in used dataset). Through the issues identified, it is very evident that dataset size has a direct impact on the computational cost of these solutions. Hence, efficiency of these solutions may vary with size of dataset used for experimentation. Figure 2a and b shows that the solution proposed by Wu et al. [5] has been examined with the highest number of dataset attributes, i.e., 25, followed by Kesarwani et al. [18], Park et al. [26], Du et al. [24], Samanthula et al. [3] and Liu et al. [11] using 23, 22, 18, 6 and 6 dataset attributes, respectively. The dataset used by Kesarwani et al. [18] contained 30000 instances.

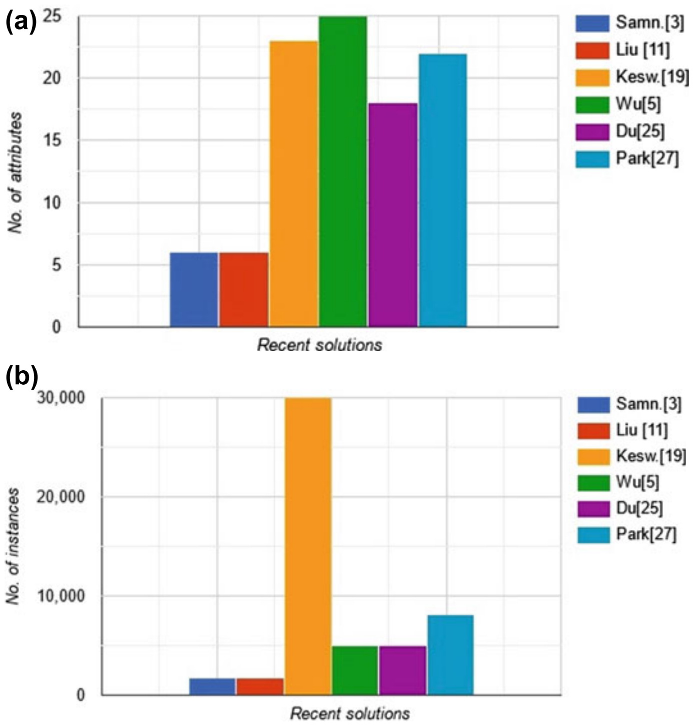


Fig. 2 **a** No. of attributes used in dataset by recent state-of-the-art solutions, **b** No. of instances used in dataset by recent state-of-the-art solutions

Park et al. [26] used 8124 instances, Samanthula et al. [3] and Liu et al. [11] used 1728 instances, whereas Wu et al. [5] and Du et al. [24] used 5000 instances for experimentation. Real-life datasets can be even larger in size than what is considered for study in [5] and [18] since they are generally populated with continuous real-time values. The performance of these existing solutions with such large datasets is highly debatable.

Secondly, we evaluated these solutions based on the run-time or execution time required by each of these solutions to generate the outcome in privacy preserved way. Figure 3 describes the run-time incurred by each of these solutions. The solution proposed by Du et al. [24] incurs least run-time, i.e., 0.614 min, when the number of attributes considered is 18, the number of data instances is 5000 and the number of nearest neighbors considered is 9. However, it is observed that the accuracy of Du et al.'s solution [24] reduces significantly when the number of attributes considered is 18 or more. This shows that the attribute size not only affects the computational cost of privacy preserved solutions but also the accuracy.

Figure 4a shows the effect of increased number of neighbors (i.e., k) on the run-time, also Fig. 4b is the *log* version of the same for better understanding of this effect. By observation, we understand that as the considered number of nearest neighbors

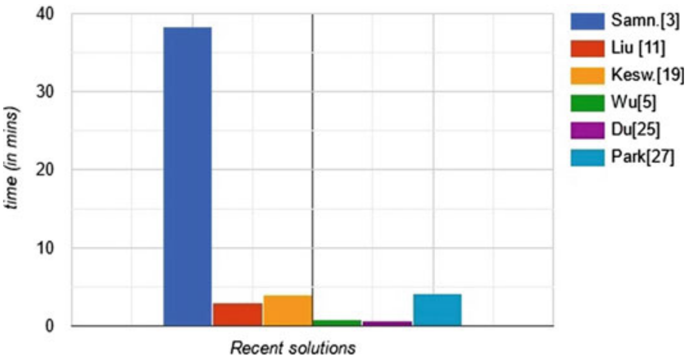


Fig. 3 Run-time required by recent state-of-the-art solutions

increases, the run-time also tends to increase. However, the solution proposed by Park et al. [26] has constant run-time (i.e., 4.16 min) for values of k due to its parallel execution approach.

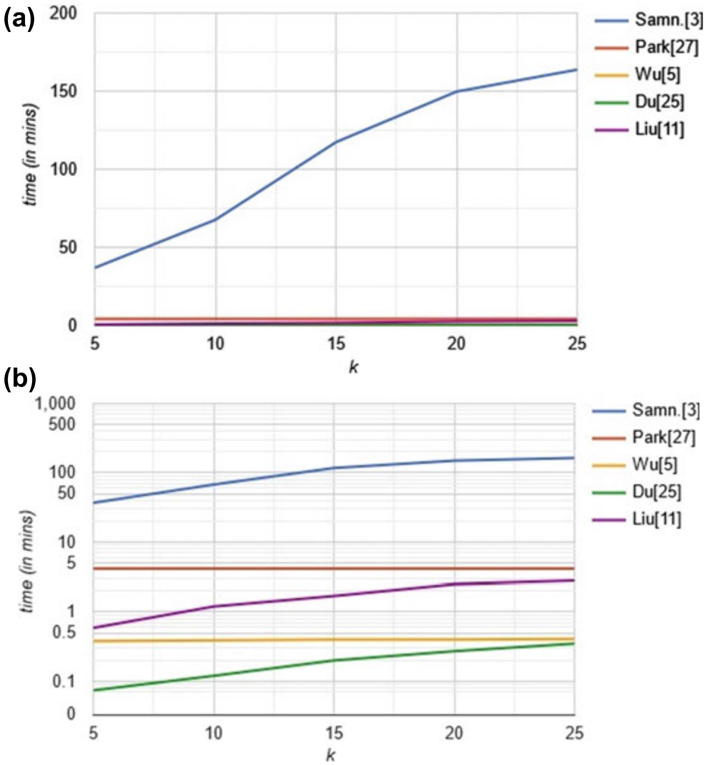


Fig. 4 **a** No. of nearest neighbors versus run-time, **b** No. of nearest neighbors versus *log* of run-time

6 Conclusion

The most important and recent issue in this research area is that even if all the privacy requirements of an outsourced *kNN* classification problem as stated in [3] are satisfied, still the solutions induce heavy computational overhead making them not enough practical in real-world scenarios. Performance of the existing solutions under scaled datasets is also an issue, and even the most efficient solutions to the outsourced *kNN* classification problem till date collapse as the dataset is scaled both attribute-wise and size-wise and their performance reduces significantly.

So in this regard, this research area of privacy preserved classification has huge scope for exploration so as to device efficient solutions in the near future. Research in this direction will be a great attempt to serve a dual purpose of providing the users with the advantages of outsourced *kNN* classification and also preserving their privacy while doing the same in an efficient way so that these privacy preserving solutions can be deployed in real-world scenario.

References

1. Gaikwad VS, Walse KH, Thakare VM (2020) Review of the state-of-the-art methods for privacy preserved classification in outsourced environment. In: International conference on innovative trends in information technology, ICITIIT 2020, Kottayam, India, 13–14 February 2020, pp 1–6
2. Elmehdwi Y, Samanthula BK, Jiang W (2014) Secure k-nearest neighbor query over encrypted data in outsourced environments. In: IEEE 30th international conference on data engineering, ICDE 2014, Chicago, IL, USA, 31 March–4 April 2014, pp 664–675
3. Samanthula BK, Elmehdwi Y, Jiang W (2015) k-Nearest neighbor classification over semantically secure encrypted relational data. IEEE Trans Knowl Data Eng 27(5):1261–1273
4. Wu W, Liu J, Rong H, Wang H, Xian M (2018) Efficient k-nearest neighbor classification over semantically secure hybrid encrypted cloud database. IEEE Access 6:41771–41784
5. Wu W, Parampalli U, Liu J, Xian M (2019) Privacy preserving k-nearest neighbor classification over encrypted database in outsourced cloud environments. World Wide Web 22(1):101–123
6. Khodaparast F, Sheikhalishahi M, Haghighi H, Martinelli F (2018) Privacy preserving random decision tree classification over horizontally and vertically partitioned data. In: IEEE 16th Intl conf on dependable, autonomic and secure computing, Athens, pp 600–607
7. Liang J, Qin Z, Ni J, Lin X, Shen X (2019) Efficient and privacy-preserving outsourced svm classification in public cloud. In: IEEE international conference on communications (ICC), Shanghai, China, pp 1–6
8. Li T, Huang Z, Li P, Liu Z, Jia C (2018) Outsourced privacy-preserving classification service over encrypted data. J Netw Comput Appl 106:100–110
9. Yang H, He W, Li J, Li H (2019) Efficient and secure *kNN* classification over encrypted data using vector homomorphic encryption. In: ieee international conference on communications (ICC), pp 1–7
10. Cheng K, Wang L, Shen Y, Wang H, Wang Y, Jiang X, Zhong H (2017) Secure k-NN query on encrypted cloud data with multiple keys. IEEE Trans Big Data
11. Liu L, Su J, Chen XR, Huang K, Deng RH, Wang X (2019) Toward highly secure yet efficient *KNN* classification scheme on outsourced cloud data. IEEE Internet Things J 6(6):9841–9852
12. Paillier P (1999) Public key cryptosystems based on composite degree residuosity classes. Eurocrypt, pp 223–238

13. Elgamal T (1985) A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans Inf Theory 31(4):469–472
14. Gentry G (2009) Fully homomorphic encryption using ideal lattices. In: ACM STOC, pp 169–178
15. Kerschbaum F (2015) Frequency-hiding order-preserving encryption. In: Proceedings of the 22nd ACM SIGSAC conference on computer and communications security, Denver, USA, pp 656–667
16. Liang J, Qin Z, Xiao S, Zhang J, Yin H, Li K (2018) MPOPE: Multiprovider order-preserving encryption for cloud data privacy. In: Proceedings of secure communication Springer 238, pp 808–822
17. Chai Y, Zhan Y, Wang B, Ping Y, Zhang Z (2020) Improvement on a privacy-preserving outsourced classification protocol over encrypted data. In: Wireless networks, Springer, 26, pp. 4363–4374
18. Kesarwani M et al (2018) Efficient secure k-nearest neighbours over encrypted data. In: Proceedings of the 21th international conference on extending database technology, EDBT 2018, Vienna, Austria, 26–29 March 2018, pp 564–575
19. Murthy S, Vivek S (2019) Cryptanalysis of a protocol for efficient sorting on SHE encrypted data: In: cryptography and coding (IMACC). Lecture notes in computer science, vol 11929, pp 278–294. Springer, Cham, Switzerland
20. Bogos S, Gaspoz J, Vaudenay S (2018) Cryptanalysis of a homomorphic encryption scheme. Cryptogr Commun 10(1):27–39
21. Oktay KY, Kantarcioglu M, Mehrotra S (2017) Secure and efficient query processing over hybrid clouds. In: IEEE 33rd international conference on data engineering (ICDE), San Diego, pp 733–744
22. Dautrich JL, Ravishankar CV (2012) Security limitations of using secret sharing for data outsourcing. In: Data and applications security and privacy XXVI. DBSec 2012, vol 7371, pp 145–160
23. Hsu YC, Hsueh CH, Wu JL (2020) A privacy preserving cloud-based k-NN search scheme with lightweight user loads. Computers 9(1)
24. Du J, Bian F (2020) A privacy-preserving and efficient k-nearest neighbor query and classification scheme based on k-dimensional tree for outsourced data. IEEE Access 8:69333–69345
25. Zheng Y, Lu R, Shao J (2019) Achieving efficient and privacy-preserving k-NN Query for outsourced eHealthcare data. J Med Syst 43
26. Park J, Lee DH (2020) Parallelly running k-nearest neighbor classification over semantically secure encrypted data in outsourced environments. IEEE Access 8:64617–64633
27. Kim HI, Kim HJ, Chang JW (2019) A secure kNN query processing algorithm using homomorphic encryption on outsourced database. In: Data & knowledge engineering 123
28. Lian H, Qiu W, Yan D, Huang Z, Tang P (2020) Efficient and secure k-nearest neighbor query on outsourced data. In: Peer-to-peer networking and applications, Springer Nature
29. Liu L, Chen R, Liu X, Su J, Qiao L (2020) Towards practical privacy-preserving decision tree training and evaluation in the cloud. IEEE Trans Inf Forensics Secur 15:2914–2929
30. Jain P, Gyanchandani M, Khare N (2018) Differential privacy: its technological prescriptive using big data. J Big Data 5(15), Springer

Workload Aware Cost-Based Partial Loading of Raw Data for Limited Storage Resources

Mayank Patel, Nitish Yadav, and Minal Bhise

Abstract Modern-day applications generate a large amount of data stored in raw formats. The traditional way of data processing requires the entire raw data files to be loaded into a database management system DBMS. Although traditional DBMS offers optimized query execution on loaded data, the loading process itself is time-consuming. On the other hand, raw data query processing engines suffer from slow query execution times QET even though loading time gets minimized. The paper presents Workload and Storage Aware Cost-based raw data partitioning WSAC technique to improve the QET of frequent queries by loading only target data into the DBMS. The technique considers query workload, storage budget, and loading cost of attributes. WSAC outputs three partitions: (1) Memory budget partition, (2) Remaining workload partition, (3) Unused data partition of the dataset table. Only the first partition is loaded in the DBMS, and two other partitions stay in raw format. The technique is demonstrated using the Sloan Digital Sky Survey SDSS dataset. WSAC improves the storage budget utilization by 5%. It also reduces the workload execution time WET by 95.37% as compared to the original.

Keywords Cost function · Large datasets · Partitioning technique · Raw data · Resource utilization · Storage budget · WSAC algorithm

1 Introduction

Modern-day applications datasets like Smart Home Datasets SHD [1], Sloan Digital Sky Survey SDSS [2], and Large Hadron Collider LHC [3] contain a large amount of

M. Patel (✉) · N. Yadav · M. Bhise
Distributed Database Group, DA-IICT, Gandhinagar, India
e-mail: mayank@daiict.ac.in

N. Yadav
e-mail: 201911035@daiict.ac.in

M. Bhise
e-mail: minal_bhise@daiict.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_74

data stored in raw formats. These applications need to analyze the massive amount of data for knowledge discovery, error solving, or finding the trends to provide users with a personalized experience. The network logs, system logs, page access logs, page click events, and other events data are stored in raw files for later analysis. IoT applications also use raw files to collect streaming sensor data to avoid high loading time [4]. The data stored in raw files also facilitate faster loading of data into databases with bulk loading techniques.

The traditional way of raw data processing requires the entire raw file to be loaded into the database. The execution of queries on the data is possible once the entire dataset is parsed, tokenized, and converted into their respective data types in the database. Furthermore, indexes need to be created after or during the data loading process for faster query processing. In comparison, raw data processing engines allow querying the raw files directly. These engines eliminate data loading and indexing costs before the arrival of queries. Raw engines are being developed to extract only necessary data from raw files needed by the query [5]. It saves a lot of time by not processing unwanted data. However, when a query arrives, it needs to access the required data from the raw file. The late processing of raw data increases the query execution time QET due to parsing, tokenizing, and data type conversion operations on required data. Most raw engines keep these parsed data in the main memory to improve the QET of future queries [6]. The main memory cached data gets removed whenever the raw file gets modified, or new data needs to be cached. Future queries may require old data that was removed from the cache. This raises the issue of reparsing the same data as the data needs to be re-accessed from the raw file. The reparsing problem increases the workload execution time and takes a lot of resources, increasing system operation costs.

Problem Statement: The existing in-situ engines, stream processing engines, DBMS, and main memory databases can efficiently process the data existing in the main memory. However, the main memory size is limited. Research has shown that only 8% of hot data can answer 64% of queries while improving QET by 84% [7]. The challenge is identifying which part of the dataset is most important for caching or loading in the database for limited memory or storage budgets that cover most workload queries.

2 Related Work

Most in-situ engines cleared the processed data from the main memory after the completion of query execution which elevated reparsing issues. The raw data query processing techniques were proposed, caching and indexing the data into the main memory for faster response times to resolve the reparsing issue [1, 6]. The invisible loading technique loads the data parsed by the query into the database before flushing the memory to create space for new data [8]. This technique incrementally loads the attributes into the database whenever a query accesses a new attribute, which is slower than upfront complete loading. This technique loads the data even if an attribute is accessed only once during the entire workload execution. In worst cases, it may culminate in loading all the data into the database. To avoid such issues, researchers have imposed database storage limitations to reduce the data loading. The techniques require workload information to decide which data are frequently accessed and best suited for loading [9]. The resource monitoring showed that real-time loading of data impacted the QET of running queries due to IO waiting. The delayed loading of parsed data was recommended centered on resource availability [10].

The traditional partitioning techniques like horizontal partitioning, vertical partitioning, or hybrid partitioning [11] can be applied using relational approaches to partition the relational data considering workload analysis. Similarly, semantic-aware partitions are created using graph approaches [12]. The query execution is faster when complete data used by the query is loaded as the query uses processed data. The query plans can be made easier by knowing available structures and indices already built on the data. However, these partitioning techniques take too much time in partitioning and rearranging the data on the database level compared to raw data and partial loading techniques [1, 9]. Researchers have tried to implement client-assisted partial loading to utilize client CPU cycles to reduce server load [13]. But it only works on string data and builds light indices based on predicates provided by the server. The delayed index creation techniques build the indexes on the data used by the queries at the cost of a slightly slower query response time for initial queries [14, 15].

The techniques discussed earlier in this section considered processing costs, storage budget, or main-memory for partial loading of the data. However, most techniques require manual involvement to provide storage limitations for different applications. How can one decide how much storage budget would be needed for a given machine? How many attributes can be fitted in the given storage budget? The paper proposed to monitor the availability of main memory resources to decide the storage budget. The number of attributes that can be fit in the given storage budget gets calculated using a cost function. The cost function considered sample dataset to find actual attribute storage requirements on server-side databases. The following section proposes a solution storing a small part of data in a database to reduce the data loading time. The in-memory indexes are built on the remaining raw data if needed to improve the QET.

3 Workload and Storage Aware Cost-Based Technique WSAC

Row store databases and raw data processing engines need to read entire records stored inside a database or a file to answer the queries which require those specific records. Data partitioning techniques can improve the QET time by reducing the accessed data. For traditional databases, partitioning and re-partitioning is very costly. Whereas partitioning raw data is relatively less costly when considering frequent changes in the workload [1]. This section discusses the proposed Workload and

Table 1 WSAC data structure examples

#s_d	Table_list	Attr_list
1	photoprimary	objid
		run,
		ra
		rerun,
		camcol
		field
		obj
		…

(a) Schema Dictionary

#que_d	Q_ID	Table_list	Attr_list
1	1	photoprimary	objid
			run
			rerun
			…
2	2	photoprimary	objid
			type
			flags_r
			…

(b) Nested Dictionary of Queries

Storage aware Cost-based technique WSAC, which considers the query workload, read–write costs, and storage parameters to decide the optimal partitions of raw data files. Most algorithms provide two partitions of a table to distribute among database and raw engines [9]. The first partition is to be loaded in a database, and the other one is to be kept in raw format. However, the proposed technique suggests three partitions for big tables. The third partition is a list of attributes that do not get used by the workload and should be kept raw.

3.1 WSAC Data Structures

Most techniques require the basic arrays, list data structures. The proposed technique required 2D or more complex array data structures. We had used python dictionary and nested dictionary data structure during implementation. Python uses hash functions to map keys to values for dictionaries. Table 1 shows examples that are used in sub algorithms. The s_d dictionary shown in Table 1 (a) stored a list of attributes associated with tables. The instance of *photoprimary* table from the SDSS dataset has been displayed for understanding. Table 1 (b) represents que_d, the nested dictionary of workload queries that contain tables and attributes used in each query.

3.2 WSAC Algorithm

This section describes sub-functions used by Workload and Storage aware Cost-based Technique WSAC; (1) Attributes and Entities Extraction (2) Cost Function, (3) Query Coverage QC, (4) Attribute Usage Frequency AUF. After extracting attributes, the cost function finds the access costs of workload attributes. The QC sub-algorithm tries to cover all attributes of frequent queries for a given storage budget, while AUF tries to fill up the remaining budget with the most frequent attributes.

Attributes and Entities Extraction. This function extracts table names, attributes names used in the workload queries, and the original dataset schema. The database schema file is provided in a data definition language DDL format. After removing additional words and data types, a dictionary *s_d* of entities and their attributes gets generated. From the provided workload file, *que_d* gets populated, which lists tables and attributes used by each query. Comparing both the dictionary lists provides us with two raw data partitions *wa_l,* and *s_d–wa_l*. The *wa_l* is the attributes used by workload, and the *s_d–wa_l* lists the unused attributes not used by the given workload.

Cost Function. The attribute and entities extraction part provides the list of attributes and tables used by the workload queries. The algorithm calculates the cost of each attribute used by query workload listed in *wa_l* by performing operations on the actual raw data. The cost function extracts each attribute from the raw data file and loads it into the given database system to find the attribute's extraction time, load time, and size. The values get recorded in the *cost_d* dictionary list. The cost of an *i*th attribute is:

$$Cost\_d_i = (\text{Read} - \text{Extract Time } r_i + \text{Load Time } l_i) \qquad (1)$$

The above equation calculates the cost of an attribute *i*. Read-Extract time represents the time required to read the raw file and extract the given attribute. The load time represents the time taken to load the attribute data into a database as a single-column table. The size of attribute $\mathbf{S}_i$ is queried from the database by checking the actual size of the single-column table on the storage medium, which can be HDD, SSD, or RAM.

1 Cost Function

```
s_d = Schema Dictionary; wa_l = Workload Attributes List;
que_d = Dictionary of Queries; cost_d = read/write &
storage costs of attributes;
```

```
def CostCalculation(s_d, wa_l, que_d, dataset CSV):
 for query i in que_d:
   for each attribute j in que_d[i]:
     for each attribute A in wa_l:
         If j = A
           Find index k of A in s_d:
           cost_d[A].r = readfromraw(j, k, dataset CSV)
           cost_d[A].l = loadattribute(j,k.lower())
           cost_d[A].S = size of A in loaded format
 retrun cost_d;
```

Storage Budget. When processing a big amount of data, the word big can be related to many parameters of data processing requirements like the size of data, type of operations that need to be performed, and maximum wait time affordable to get results for that application. The algorithm uses the parameters storage budget *B,* which indicates cache size or database size, which a machine can process efficiently. WSAC tries to reduce the total workload execution time defined in Eq. 2 by covering attributes of queries that can fit in the available storage space *B*. The total Workload Execution Time is:

$$\mathrm{WET} = \sum_{i=1}^{m} \mathrm{DLT}(A_i) + \sum_{j=1}^{n} \mathrm{QET}\big(Q_j\big) \tag{2}$$

A research paper discussing a storage-aware partitioning algorithm directly assumed budget *B* as some *x* number of attributes [9]. While the proposed algorithm first takes an actual storage budget *B* in MB and finds out how many attributes can be covered in the given budget? In the earlier case [9], if the storage budget is given as $B = 3$ attribes. Then the algorithm may cover three attributes having datatype char(100) or three boolean attributes. It means the storage budget size may differ significantly in each case for covered attributes. On the other hand, the proposed algorithm checks the actual storage size used by the attributes considering the actual datatype and data values, which allows us to fill the available storage budget more accurately.

Workload Awareness. The WSAC uses two sub-algorithms, Query Coverage QC and Attribute Usage Frequency AUF to integrate workload awareness.

Query Coverage QC. The function uses a modified query coverage algorithm [9] to find the queries that can fit inside the given storage budget *B*. The function uses the storage cost values of each attribute recorded in *cost_d* for each query *que_d* and adds them. The algorithm tries to cover most queries from the frequent queries set based on the storage budget. The list of queries gets filtered based on their required storage budget requirements. The queries requiring less budget than given B are considered further. The first query is chosen based on storage cost from the most frequent query set. The attributes of the covered query get added in the covered attributes list *ca_l* and the covered query in the *cq_l*. Now, the remaining queries having covered attributes will need a lower storage budget to get covered. This way, each query is checked if it can be covered within a given storage budget until no other query can be covered.

2 Query Coverage - QC

```
B = Storage budget B in MB; ca_l = List of covered Attributes; cq_l = List of covered Queries;
```

```
def QueryCoverage(que_d, cost_d, B):
  ca_l=0,cq_l=0   #list of covered attributes & Queries
  For each query q from freq. query set que_d
       if (SUM(size of attributes of q[i])) < B :
       for each attribute A in q[i]
           If A is not in ca_l
             if size of A < remaining Budget B
               Add A in ca_l list & update B
         Add q in cq_l if all attributes are in ca_l
       Else
         Query q cannot be covered.
return ca_l, cq_l;
```

Attribute Usage Frequency AUF. This sub-function attempts to fill up the remaining storage budget using the remaining attributes used by uncovered queries. The list of remaining attributes gets extracted from the remaining queries with the frequency and removing covered attributes. The attribute coming in most queries will be at the top of the list. This way, the algorithm tries to reduce the query execution time for partially covered queries. If two attributes have the same frequency, then the attribute having a high cost is considered. The final list of covered attributes *ca_l* for each table decides the remaining two partitions. All covered attributes *ca_l* covered in the given storage budget will get loaded into a database, so all the covered queries *cq_l* can be answered directly from a database. The raw partition is created for attributes used in the workload, but the algorithm could not cover them due to storage limitations.

The final output of WSAC consists of three vertical partitions for the given table; (1) Memory budget partition *ca_l*, (2) Remaining workload partition *(wa_l-ca_l)*, (3) Unused data partition *(s_d-wa_l)*. If any attributes exist in the list (*wa_l-ca_l*), then it means some data needs to be fetched from raw partitions to answer partially covered or non-covered queries. If the storage budget is sufficient to store all the attributes existing in list *wa_l*, then there will be only two partitions *ca_l,* and (*s_d-wa_l*). The loaded data will cover all the queries.

3 Attribute Usage Frequency - AUF

```
wa_rl = list of remaining non-covered attributes
def AUF (ca_l, cost_d, B, s_d, wa_l):
 wa_rl= wa_l-ca_l
 wa_rl= Sort(wa_rl) #based on Attribute Freq.
 For each attribute A from wa_rl
      if $(\mathrm{Cost\_d}[A].S) < B$ :
        if(A.Freq. == (A+1).Freq.)
          if Cost_d[A].r+Cost_d[A].l >
                               Cost_d[A+1].r+Cost_d[A+1].l
            Add A in ca_l list & update B
            #Cover high Read/Load Cost Attributes First
          Else if $(\text{Size of }(A + 1)) < B$ :
              Add (A+1) in ca_l list & update B
      Else
        Attribute cannot be covered.
return partitions ca_l, (wa_l-ca_l), (s_d-wa_l);
```

4 Experimental Setup

This section describes the experimental setup consisting of Hardware and Software Setup, Dataset and Query Set, and Experiment Flow.

4.1 Hardware and Software Setup

The hardware configuration of the machine included a quad-core Intel i5-6500 CPU clocked at 3.20 GHz. It had 16 GB of RAM and Intel HD Graphics 530 (Skylake GT2) in-built, running 64-bit Ubuntu 18.04 LTS operating system. A SATA hard disk drive had 500 GB of space and 7200RPM rotation speed is used as a permanent storage medium. The software required to run the WSAC algorithm is Jupyter running python code. The raw data query processing framework is used handle the raw data [16]. The database management system PostgreSQL is used by framework to query raw data after loading into a database. While the framework uses PostgresRAW extension to execute queries directly on raw files. The *top* command provided the real-time data of RAM utilization.

4.2 Dataset and Query Set

The dataset used for experiments is a real-world astronomical dataset of stars, galaxies, and other sky objects called Sloan Digital Sky Survey SDSS. The size of the recent data release named DR-16 is 273 TB [2]. From version DR-16 of SDSS 16 GB of *PhotoPrimary* data which answered the majority of queries, was extracted and used for the experiments. The number of records in the extracted table is 4 M. The top 1000 frequent *PhotoPrimary* view queries are used for the experiment. These queries represented 51% of DR-16 workload. The similar queries have been grouped based on the similarity of the attributes and query type, forming 12 query groups.

4.3 Experiment Flow

The algorithm flow has been discussed in detail in Sect. 3. Therefore, this section will discuss the parameters and files that are given as input to the sub-algorithms, as shown in Fig. 1. First, the database schema—the DDL files of *PhotoPrimary* table having 509 attributes, and the frequent query set files are provided for the initial attribute and entity extraction phase. The input for the cost calculation phase is 1 M records of the *PhotoPrimary* table in CSV format. The cost calculation function calculated the read-extraction, load cost and size for attributes used by the workload. Once the attribute is loaded in a single-column table, the function checks the actual table size S_i required to store the attribute *i*. QC and AUF sub-algorithms of WSAC then use the output costs to find coved attributes and queries. The partition consisting of covered attributes is loaded in PostgreSQL. The raw partition files are linked to tables using the PostgresRAW extension configuration files. Now, the java code executes the covered queries on PostgreSQL and partially covered queries on the combination of PostgreSQL and PostgresRAW. The database and raw partitioned are joined using the *ObjID* Primary key *PK_A*. The 12 queries from the frequent query set were executed four times, and the average is considered for accurately counting the QET time. If any new queries come that don't have any covered attributes, they can be answered using the 3rd raw partition.

5 Results and Discussion

This section discusses the results of partial loading experiments based on partitions provided by the WSAC technique.

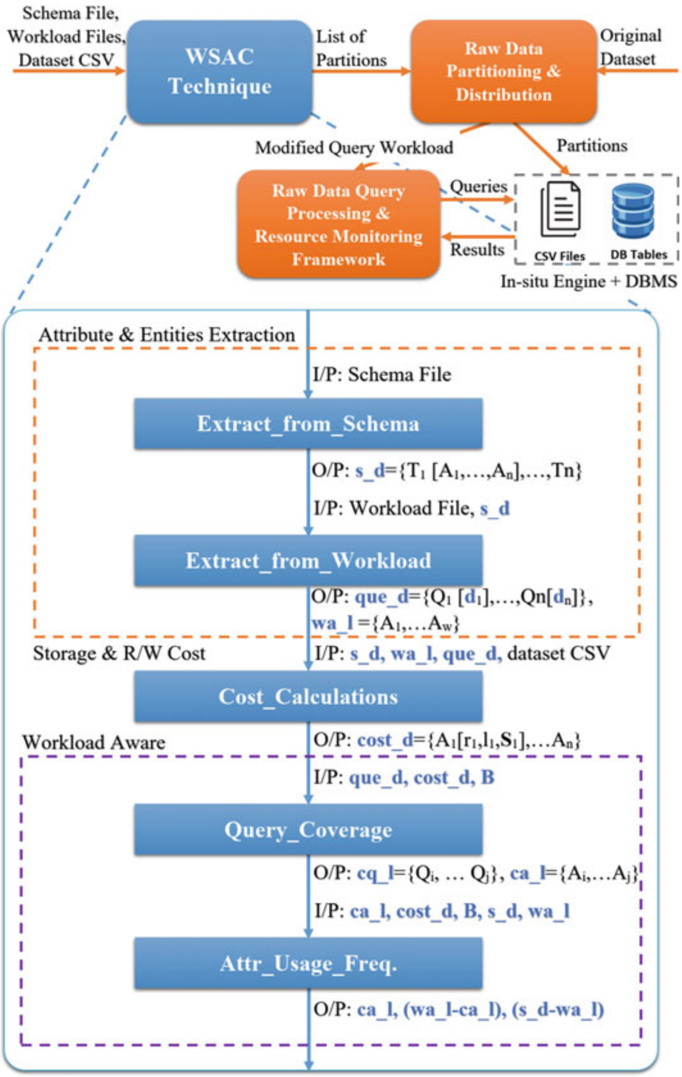
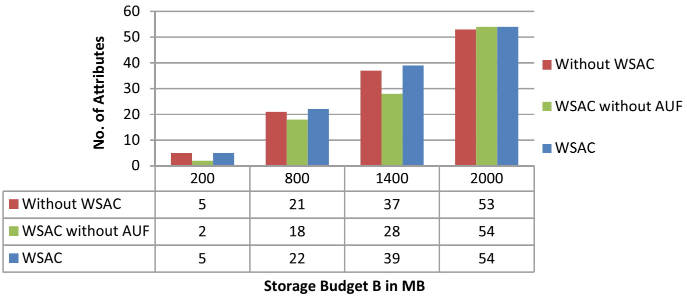

Schema File, Workload Files, Dataset CSV
WSAC Technique
List of Partitions
Raw Data Partitioning & Distribution
Original Dataset
Modified Query Workload
Partitions
Raw Data Query Processing & Resource Monitoring Framework
Queries
Results
CSV Files
DB Tables
In-situ Engine + DBMS
Attribute & Entities Extraction
I/P: Schema File
Extract_from_Schema
O/P: s_d=$\{T_1 [A_1,\ldots,A_n],\ldots,Tn\}$
I/P: Workload File, s_d
Extract_from_Workload
O/P: que_d=$\{Q_1 [d_1],\ldots,Qn[d_n]\}$,
wa_l =$\{A_1,\ldots A_w\}$
Storage & R/W Cost
I/P: s_d, wa_l, que_d, dataset CSV
Cost_Calculations
O/P: cost_d=$\{A_1[r_1,l_1,S_1],\ldots A_n\}$
Workload Aware
I/P: que_d, cost_d, B
Query_Coverage
O/P: cq_l=$\{Q_i, \ldots Q_j\}$, ca_l=$\{A_i,\ldots A_j\}$
I/P: ca_l, cost_d, B, s_d, wa_l
Attr_Usage_Freq.
O/P: ca_l, (wa_l-ca_l), (s_d-wa_l)

Fig. 1 Experiment flow for WSAC technique

5.1 Storage Resource Utilization

Figure 2 compares attributes covered by the algorithm choosing a static number of attributes as storage budget [9] with WSAC. The red bar shows fixed attributes as a storage budget named *without WSAC*. The green bar represents *WSAC without AUF,* which used only Query Coverage QC sub-algorithm, and the blue bar shows attributes covered when WSAC used QC and AUF sub-algorithms for different storage budget



	200	800	1400	2000
🟥 Without WSAC	5	21	37	53
🟩 WSAC without AUF	2	18	28	54
🟦 WSAC	5	22	39	54

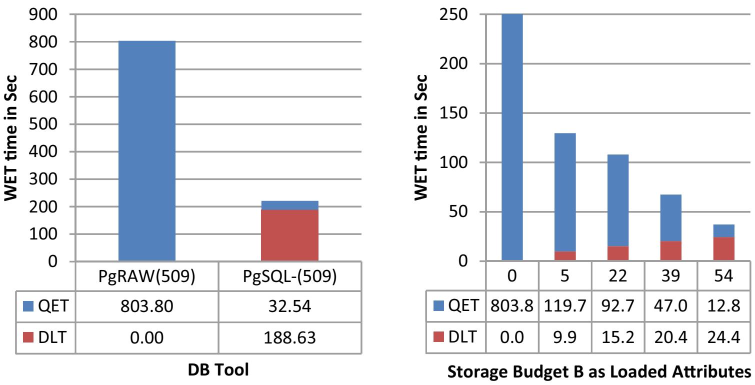
Fig. 2 Storage utilization as attributes covered by WSAC technique

options. We assumed that *without WSAC,* the covered attributes might have been calculated based on the average size of attributes that can fit in a given storage budget B. The average size of attributes for *PhotoPrimary* table columns is 38.41 MB. That means *without WSAC* average number of attributes covered for 200, 800, 1400, and 2000 MB storage budget B would be 5, 21, 37, and 53. The number of queries covered by WSAC algorithms is 1, 5, 9, 12 for a corresponding storage budget of 200, 800, 1400, and 2000 MB. It is crucial to cover all query attributes to get results using only the loaded part of the dataset to improve the QET. There is no guarantee that most frequent attributes would cover queries of the frequent query set in *Without WSAC* cases where QC is not a sub-algorithm. *WSAC without AUF* covered 2, 18, 28, 54 attributes, while WSAC covered 5, 22, 39, 54 attributes for the given storage budget.

The results show that the *WSAC without AUF* algorithm alone could not utilize all the available storage budget. The WSAC's AUF sub-algorithm tries to utilize the remaining space with attributes having lower storage budget requirements in the end to improve storage budget utilization. The WSAC improved the storage utilization by 20–60% compared to *WSAC without AUF*. For the *PhotoPrimary* table, WSAC technique covered 5% more attributes than the algorithms choosing average values to decide storage budget represented as *without WSAC*. When stored in a database, most columns in the *PhotoPrimary* table had *long int* or *real* datatypes having similar column sizes due to the 4-byte storage space requirement on disk. The WSAC can cover more attributes if the dataset consists *boolean* or *char(1)* datatypes that require only 1-byte storage space to store one tuple.

5.2 *Partial Loading*

The comparison of PostgreSQL PgSQL with PostgresRAW PgRAW is plotted in Fig. 3a to get the actual workload execution time required by the original non-partitioned dataset having 509 attributes for both systems independently. It can be observed that raw data query processing engines like PostgresRAW have zero data loading time. At the same time, PostgreSQL required 188 s to load the data existing



a) Without WSAC **b)** WSAC with given Storage Budget B

Fig. 3 Total workload execution time WET for SDSS dataset

in a CSV file using the fastest loading method COPY. However, PostgreSQL required 4.05% time to complete the query execution of 12 queries on loaded data compared to PostgresRAW. One thing to note is that PostgresRAW is an open-source tool and developed by a small group of researchers. Therefore it is not entirely created with industry standards nor optimized to execute JOIN queries as effectively as PostgreSQL. The PostgresRAW required 11 GB of RAM to run queries on 4.6 GB of raw data, limiting the data scaling as it would have triggered swapping and increased WET.

Figure 3b compares total workload execution time (Eq. 2) before and after the proposed WSAC technique partitioned the *PhotoPrimary* table into three partitions from which workload queries are using only two partitions. The first bar in the result shows the QET time when zero attributes are loaded into the database, which means all the queries are executed using PostgresRAW. The second bar displays the WET when only five attributes are loaded into the database. It reduced the workload execution time by 83.88%, with just one query covered by PostgreSQL. This happens because the main primary key column *objID* is loaded in the database, which helps in reducing the time required to join records. The raw files are only accessed to get the data to be projected. The 22 and 39 attributes are loaded into the database based on the storage budget, which reduced the overall workload execution time by 86.57% and 91.61% as compared to zero loaded attribute workload. The last bar shows the WET time when all attributes are loaded into the database. The 54 loaded attributes reduced the workload execution time by 95.37% as compared to zero loaded attributes. It can be observed that for 54 attribute partition, the DLT and QET time gets reduced by 87.06% and 60.66% as compared to the original database loaded in PostgreSQL. The created partitions can be cached into the main memory to reduce QET further.

6 Conclusion

The paper proposed a workload and storage aware cost-based technique WSAC for large datasets. The WSAC considered storage cost and read–write costs of attributes to decide the partitions for a given storage budget. The storage budget utilization of the WSAC technique improved by 5% compared to the average attributes covered and 20–60% compared to *WSAC without AUF*. The output partitions reduce the WET by more than 80% for different storage budgets. The efficient execution of JOIN queries is achieved using 1% loaded attributes while most projected attributes got fetched from raw format. The WET is reduced by 95.37% when all required attributes are covered in PostgreSQL compared to no loaded attributes in PostgresRAW. The WSAC technique can be applied to all types of large datasets requiring only part of the data to answer most queries. Future works can optimize cost calculations and storage budget utilization functions considering the disk block size and data compression ratio.

References

1. Olma M, Karpathiotakis M, Alagiannis I, Athanassoulis M, Ailamaki A (2020) Adaptive partitioning and indexing for in situ query processing. VLDB J 29:569–591
2. DR16-Data Volume Table | SDSS. https://www.sdss.org/dr16/data_access/volume/
3. Ailamaki A (2011) Managing scientific data. In: Proceedings of the 2011 international conference on Management of data - SIGMOD '11. p 1045. ACM Press, New York, NY, USA
4. Gorenflo C, Golab L, Keshav S (2017) Managing Sensor Data Streams. In: Proceedings of the 29th International Conference on Scientific and Statistical Database Management. pp 1–11. ACM, New York, NY, USA
5. Palkar S, Abuzaid F, Bailis P, Zaharia M (2018) Filter before you parse. Proceedings of the VLDB Endowment. 11:1576–1589
6. Alagiannis I, Borovica R, Branco M, Idreos S, Ailamaki A (2012) NoDB in action. Proceedings of the VLDB Endowment. 5:1942–1945
7. Jain A, Padiya T, Bhise M (2017) Log Based Method for Faster IoT Queries. In: IEEE Region 10 Symposium (TENSYMP), pp 1–4
8. Abouzied A, Abadi DJ, Silberschatz A (2013) Invisible loading. In: Proceedings of the 16th International Conference on Extending Database Technology—EDBT'13. pp 1–10. ACM Press, New York, NY, USA
9. Zhao W, Cheng Y, Rusu F (2015) Vertical partitioning for query processing over raw data. In: Proceedings of the 27th International Conference on Scientific and Statistical Database Management. pp 1–12. ACM, New York, NY, USA
10. Cheng Y, Rusu F (2015) SCANRAW. ACM Trans Database Syst 40:1–45
11. Padiya T, Bhise M (2017) DWAHP: workload aware hybrid partitioning and distribution of RDF data. In: Proceedings of the 21st International Database Engineering & Applications Symposium (IDEAS). pp 235–241. ACM
12. Pandat A, Gupta N, Bhise M (2021) Load balanced semantic aware distributed RDF graph. In: 25th International Database Engineering & Applications Symposium (IDEAS), pp 127–133. ACM, New York, NY, USA

13. Ding C, Tang D, Liang X, Elmore AJ, Krishnan S (2021) CIAO: an optimization framework for client-assisted data loading. In: 2021 IEEE 37th International Conference on Data Engineering (ICDE), pp 1979–1984. IEEE
14. Petraki E, Idreos S, Manegold S (2015) Holistic indexing in main-memory column-stores. In: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data. pp. 1153–1166. ACM, New York, NY, USA
15. Maroulis S, Bikakis N, Papastefanatos G, Vassiliadis P, Vassiliou Y (2021) Adaptive indexing for in-situ visual exploration and analytics. In: 23rd International Workshop on Design, Optimization, Languages and Analytical Processing of Big Data (DOLAP)
16. Patel M, Bhise M (2019) Raw data processing framework for IoT. COMSNETS. In: 10th international conference on communication systems and networks, pp 1–5

Differential Privacy Mechanisms: A State-of-the-Art Survey

Shriniwas Patil and Keyur Parmar

Abstract The advantage of digitization is the availability of enormous data that make decision-making efficient and accurate. However, the data that improve the decision-making create a wide range of privacy concerns for users. The privacy-preserving data analysis is becoming a crucial research topic in the domain of computer science. One of the popular procedure used to ensure the privacy of data is anonymization where the identifiable information related to the users are removed before using the data for analysis. However, there are several issues associated with anonymization. In this article, we discuss the differential privacy mechanisms used to ensure the privacy of data. We formally discuss the definition of differential privacy, and then provide the seminal algorithms in the domain of differential privacy that enables the privacy-preserving data analysis. We discuss the applications of differential privacy. In addition, we present the state-of-the-art issues (or research gaps) in the domain of differential privacy and provide the future research directions.

Keywords Anonymization · Differential privacy · Privacy budget · Sensitivity · Noise · Laplace

1 Introduction

In today's world, digital data are the most important, and data can be captured from almost all fields such as education, business, health care, social media, or e-commerce. The data captured from all fields influence the day-to-day life of an individual as well as enterprises. Therefore, the protection of data privacy becomes

S. Patil (✉) · K. Parmar
S. V. National Institute of Technology, Surat, India
e-mail: shriniwas1996@gmail.com

K. Parmar
e-mail: keyur@coed.svnit.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_75

impportant. The different usage of data such as the data is used to analyze for benefit of people such as drug purchase in a pandemic situation. Statistical analysis is used to release statistics, but too many and overly accurate statistics destroy the privacy. Therefore, our goal is that data analyst should not be able to learn information that are private to individuals.

There are some misconceptions among people that differential privacy [7] is the same as other data processing techniques such as data anonymization [25]. Before going deep into differential privacy [7] first, we need a clear difference between data anonymization [25] and differential privacy [7]. The data processing technique that removes or modifies individuals identifiable information is known as data anonymization [25]. The process of data anonymization happens on the company's servers. Therefore, everything depends on how much you trust. In addition, the data anonymization [25] technique is not worthy when someone has auxiliary information, and by combining auxiliary information with anonymized data, we can access the original data. These types of attacks are called a combined attack or linkage attack [30]. Authors [7] design differential privacy to overcome the dangers and to prevent unwanted public release of sensitive data that is available in a large dataset while giving insightful data of the individuals. The use of differential privacy mechanisms allows companies to collect information about the users without compromising the privacy of individuals.

In the example of a database, the goal which we are trying to achieve with a privacy-preserving database is to give access to the users to read the properties of the data as a whole and preserve the privacy of individual records [20]. The methodology is known as private data analysis, inference control, statistical disclosure control, or privacy-preserving data mining [12]. The main aim of the differential privacy [7] is that without disclosing the identities of individuals, differential privacy provides the database analyst, researchers, to gain important and insightful information from the database. The information is not only public data but some sensitive private data. Privacy is achieved by measuring how the query or request will affect individual privacy and injecting noise into the dataset.

In the case of an interconnection network, while studying or analyzing a small subnetwork, we expect to learn certain properties of the network and not the properties of the individual edges and nodes [13].

The need for security for databases is expressed by Dalenius [9] followed by Goldwasser and Micali [18] for cryptosystems as a notion of semantic security. Dwork [13] mentioned that without giving access to the statistical database no one can learn about an individual. However, privacy cannot be achieved with the mechanism because of available auxiliary information, which is the information already available with an adversary without accessing the statistical database.

The article is organized as follows. In Sect. 2, we discuss the literature review. In Sect. 3, we discuss the formal definition of differential privacy, different differential privacy mechanisms such as differential privacy with coin flipping and differential privacy with privacy guard, basic terms used in differential privacy, research gaps and challenges. Section 4 provides the results and analysis. In Sect. 5, we conclude the article with future research directions.

2 Literature Review

The differential privacy [7] is an emerging and trending topic. There is a large number of state-of-the-art literature available. The following works have been discussed in relation to differential privacy[7].

The guarantee of security certifications would today be considered inadequately broad; as present day, cryptography has molded our understanding of the risks of the leaking of partial data or information. Though we can see some studies of differential privacy [7] and some studies related to privacy in databases in the field of security, the effort seems to be negligible over the period of time. The work of Denning [10] is the closest that we can find that catches the spirit of data privacy and differential privacy [7].

For privacy-preserving data analysis, author [27] discusses that the privacy-integrated queries (PINQ) are trustworthy platforms. Without the help of an analyst, PINQ helps us to provide access to sensitive data that doesn't require privacy expertise. PINQ's behavior and the interface match with the language-integrated queries (LINQ) [4]. PINQ internally uses mathematical formulas and is built on top of the. NET framework of LINQ is available as a software bundle which permits developers and IT professionals to fetch data from any data source using privacy-enabled queries. The PINQ makes things easier as the PINQ doesn't require understanding the mathematics behind its working. PINQ works on 'request/reply model' which doesn't add noise to the results of the computation since the data are stored on a trusted data server.

The Map Reduce paradigm is used by Airavat [32] for the implementation of differential privacy. The Airavat model has many drawbacks. For example, the Airavat model cannot restrict computation performed by untrusted code. In addition, the Airavat model considers map programs as untrusted computation and reduce programs as trusted computation. The Airavat model has limited support for the reducer functions and doesn't satisfy the goal of data security as the model is vulnerable to timing and state attacks. Therefore, we take the Airavat model as a reference for implementing data privacy using map reduce for big data.

Mohan et al. [29] proposed a GUPT mechanism that impacts the block size and accuracy. The relationship between block size and noise is directly proportional. In addition, the relationship between block size and estimation is inversely proportional. We require to choose out block size optimally. In the case of Map Reduce, the block size is compared to the mapper. The impact on the overall output can be defined as the number of mappers optimally chosen.

In 2016, Apple Inc. implemented the techniques of differential privacy in their iOS 10 operating system [1]. The model uses three transformations, namely hashing using cryptography, sub-sampling, and noise. Hashing is an efficient cryptographic

Table 1 Comparative study of different differential privacy mechanisms

Model and Description	Advantages	Disadvantages
PINQ 2009 [27]—intuitive differential privacy which ensures, at runtime, that requests cling to an overall security spending plan	Guarantees privacy. Improves the portability privacy between datasets and domains	Vulnerable to privacy budget, state, and timing attacks
Airavat 2010 [32]—mandatory access control such as access control combined with differential security	Allows a variety of privacy-preserving map options with reduced calculations and scalable	Only, the map program is considered 'untrusted' Vulnerable to state and timing attacks
GUPT 2012 [29]—assures careful data examination is easy for security non-specialists. The master can move the data mining ventures and GUPT guarantees the security	Protection from side-channel attacks such as privacy budget attacks, state attacks, and timing attacks	The output dimensions are assumed to be known in advance. Inherit differential privacy constraints in terms of sharing the privacy budget
Apple's privacy 2016 [1]—collects the data on what websites are being used, what types of emojis are used, and what words people uses frequently	Privatize the information using local differential privacy on the user's device. Therefore, Apple's servers do not get the user's original data	The local differential privacy is performed on the user device. Therefore, reduces the performance of users devices by making the use of resources
US Census 2008 [22]—uses differential privacy to protect the confidentiality of patient data and to identify driving samples	Protected against reconstruction attacks and re-identification attacks	The identity can be disclosed by matching reconstructed microdata with auxiliary information [16]
Geo privacy 2014 [2]—ensures the customer's correct location while allowing induced information to be obtained by specific organizations	Gradually degrades the user's privacy over time	Do not rely on the adversary's prior knowledge
Telco big data 2015 [19]—implemented three fundamentally different privacy architectures and deployed in the telecommunication big data platform	The use of the Markov decision process to select a tree structure in a hybridized structure improves the performance	Need a large volume of data to reduce the accuracy loss

function that gives random-looking characters as output by taking sample data as input. The random-looking data are a unique set of strings. Secondly, taking only a portion of the data is required instead of the whole dataset. In the third transformation, the noise is injected that hides the sensitive private data.

Mallya et al. [26] implemented differential privacy using the Diffie-Hellman [11, 28] and AES [8] algorithms. The authors created an application scenario for vehicular ad hoc networks (VANETs) where vehicles (nodes) travel on a specified path. There is a possibility of privacy leakage where nodes could be tracked with the help of information transmitted from node to node, and node to infrastructure. To trash the possibility of privacy leakage authors created, one differential privacy protection protocol (DPPP) based on the Diffie–Hellman key exchange algorithm [11, 28] over an insecure network and AES 256 algorithm [8] is used for encrypting the content that needs to be transmitted.

Apart from the differential privacy mechanisms described above, there are different mechanisms available for differential privacy [3, 17, 34]. The comparative study of different differential mechanisms is shown in Table 1.

3 Differential Privacy: State of the Art

There exist different techniques that create differentially private algorithms. In this section, we introduce the formal definition of differential privacy along with different techniques such as randomized response (coin flipping) and differential privacy with privacy guard. We discuss the basic terms of differential privacy. Then, we also discuss the research gaps and challenges in differential privacy.

3.1 Differential Privacy

Differential privacy [7] is not merely a calculation but a concept as introduced by Dwok, Nissim, McSherry, and Smit.

$$Pr[M(x) \epsilon S] \; \leq \; e^{\varepsilon} \; Pr[M(y) \epsilon S] \tag{1}$$

Where M gives ε-differential privacy if all pairs of datasets x and y differ in one person's data for all events S. The datasets are termed as adjacent datasets. In other words, adjacent datasets are almost identical, but one is slightly smaller than the other, and the larger one contains one more row of data [21].

Every possible output of event S has a probability. When the dataset is x is almost the same as e^{ε} multiplying by the probability, we see S when the dataset is y. We can say that the ratio of both probabilities is almost e^{ε}. We use all pairs of the dataset. Therefore, even if we swap the dataset, some inequality will be there in x and y.

Figure 1 shows how differential privacy works with randomized responses. Assume that we have reviewed the number of people by asking a question: Is your income greater than amount X or not? We store the answers on the server, but instead of storing the original answer on the server, we apply the coin flipping algorithm to add a noise. Assuming someone answer the question as 'Yes', we apply the coin flip algorithm. The coin flip algorithm generates the output heads or tails randomly. If the output of the coin flip algorithm is head, we store the original answer 'Yes' on the server. If the output of the coin flip algorithm is tail, the algorithm will run again. In

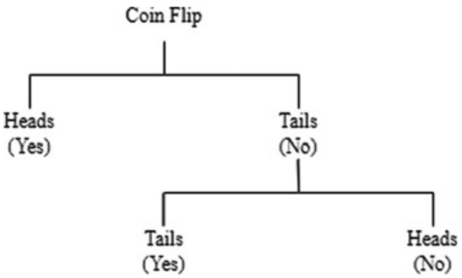


Fig. 1 Differential privacy with coin flipping

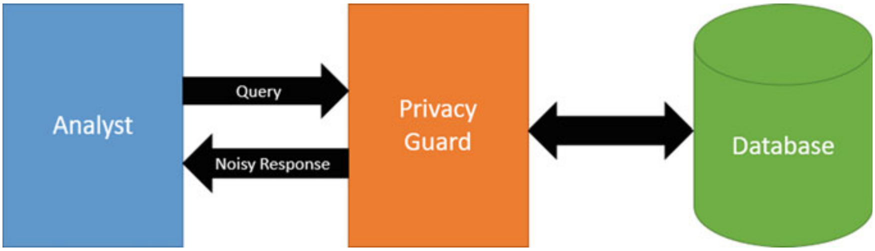


Fig. 2 Differential privacy with privacy guard [7]

the second phase, if the output of the coin flip algorithm is tail, we store the original answer 'Yes' on the server. Otherwise, we store the answer 'No' on the server. Based on the server records, we cannot get individual original data because there is at least 25% of noisy or false data stored on the server.

In another example, as shown in Fig. 2, we present the whole process of database access and study with differential privacy. Jain et al. [20] explain how the process works with the layer of privacy guard in the picture. Using a specially designed algorithm, the guard measures the impact of the privacy raised while querying the database. The guard receives a query from the user of the system, assesses the privacy, and then sends the query to the database. The database engine then returns the clean set of datasets corresponding to the query string. The data received by the privacy guard is now clean and free from any kind of noise. The privacy guard adds a noise based on the privacy impact that was defined earlier which makes data random to identify the individuality of the database records and sends back the results to the user, researcher, or analyst. Using differential privacy, the whole state of the database isn't compromised; neither affects the outcome. In the real world, algorithms use "Laplace distribution" to disperse data over a wider range and increase anonymity.

3.2 Basic Terms of Differential Privacy

We will discuss the basic terms used in the differential privacy.

Privacy Budget (ε)—the number of queries to be answered in the data available due to data privacy restrictions is decided by the privacy budget. The higher the cost of processing a query, the more private data. The ε is known as the privacy budget. The ε is a parameter used to optimize and control the output's proportion of the neighboring datasets x and y, of the function M in Eq. 1. Lower the utility, higher the security and vice-versa [21]. Normally, the value of ε is selected as 0.01, 0.1, 0.5, and 0.8. [19].

$$\varepsilon \propto \text{Accuracy}$$

Sensitivity (s)—sensitivity specifies the amount of noise to be added to the query results. Depending upon the addition or removal of a single row, the sensitivity changes. Consider Q as the counting query and x and y as neighboring datasets. The sensitivity of Q is denoted by s as mentioned below [21].

$$s = \max \mid\mid Q(x) - Q(y) \mid\mid$$

Noise (L)—the meaningless information added to every data record is mainly termed as 'noise'. The privacy budget and the sensitivity imply the noise of the differential privacy model.

$$\text{Noise}(L) = 2 * \left(\frac{s}{\varepsilon}\right)^2$$

$$a = f(D) + L \tag{2}$$

From the above Eq. 2 , the answer a is calculated by query function f, database D, and noise L, which means that we are adding some noise to the original data to preserve privacy, and thus, the adversary will not predict the original data correctly.

3.3 Differential Privacy: Research Gaps and Challenges

To the best of our knowledge, differential privacy [7] is a technique that can ensure a guarantee of privacy using both the fields of mathematics and computer technology. We have seen from the concept of differential privacy that a person has a limited effect on the published dataset if we apply one of the differential privacy mechanisms. The dominant principle here is that the person's privacy is not violated as long as that person is not included in the information gained from the dataset. However, sensitive information may be used to know about the individual's private information. The presumption means that auxiliary information with a differential private outcome to obtain correct information about the person, thereby suggesting that it is not absolute confidential to maintain differential privacy [15].

The challenge with differential privacy is to determine how to establish the privacy parameter ε. It was not sufficiently stated how to establish the right value of ε. The variable ε affects on the ability to identify a person is not clear [6]. Although differential privacy makes difficult to decide if an individual is included in a database, therefore, differential privacy makes difficult to know the record of that individual. In the normal sense, the ε parameter in ε-differential privacy does not indicate what has been exposed about the individual; rather, differential privacy restricts the effect of an individual's outcome. The incorrect value of ε causes a violation of privacy. The level of security given by the ε-differential privacy mechanism is different even for the same values of ε, depending on the values of the domain attributes and the type of queries.

The outcomes obtained from a distinctly private process will vary immensely, making them unreliable for use. There may be a major difference in the response when using the Laplace mechanism. An example is given for a differential private query for the average income of a person in the United States of America [6] with $\varepsilon = 0.25$ (respectively $\varepsilon = 1.0$). The result is ten thousand dollars, for a true value of sixteen thousand seven hundred eight dollars, and the data could be misleading [33].

The major challenge is to explore the field of real-time e-health data using differential privacy. As of now, the studies are limited to the non real-time e-health data [24].

Another challenge is regarding the usage of the differential privacy in the industries. Though a lot of studies are going on in and around the field of differential privacy, the amount of implementation is very less in comparison to the theoretical research. To implement the differential privacy mechanisms, we can relax some of the privacy guarantees.

In each level of the decision tree, the privacy budget is equally distributed which leads to an increase if the tree keeps on increasing. Comparatively, we can design a model of configurable and adjustable privacy budget assignment at each level of the tree [23].

The main research content in the field of differential privacy is the data release techniques using the theoretical basics for practical application. The fact is that there are very few comparisons and classifications between differential privacy data release techniques which indicates that there are very few achievements in the data release field. There are some new paths to be investigated. Considering the data being stored at multiple databases or multiple nodes, a challenge occurs where it's hard to guarantee the data's privacy and the user's privacy across multiple databases and nodes. There is a need for a model that is safe and efficient and requires fewer communication overheads. Given the communication costs, this creates a new challenge for privacy budgeting [22].

Another challenge is the improper disclosure of data. The basic theory of differential privacy is to take the original data, get the result from the query, and inject sufficient noise to protect individual privacy. The original/raw data available at the server may be vulnerable to different types of attacks such as malicious intrusion or insider threat. The malicious intrusion and insider threat attacks are major breaches of privacy that fall under security issues [5].

4 Results and Analysis

We analyze the differential privacy performance using a Laplacian noisy counting mechanism.

Table 2 The confidence interval results for different privacy budget values

S. No.	Privacy budget(ε)	Estimate interval	Estimate interval difference	Estimate
1	0.05	[−2.89, 6.47]	9.36	1.79
2	0.2	[1.36, 3.98]	2.62	2.67
3	0.4	[2.06, 3.62]	1.56	2.84
4	0.6	[2.40, 3.40]	1.00	2.90
5	0.8	[2.80, 3.38]	0.58	3.09
6	1.0	[2.76, 3.14]	0.38	2.95

4.1 Laplacian Noisy Counting Mechanism

To give a practical demonstration and explanation of differential privacy, we use the simulation provided by Nguyen [31] using the laplacian noisy counting mechanism. The laplacian noisy counting mechanism is one of the differential privacy algorithms, as shown in Figs. 3 and 4. The Laplace system simply computes the function and arranges each with noise drawn from the LM appropriation. The size of the noise will be changed as per the sensitivity of the function. A Laplacian mechanism is utilized when the result is numerical [21].

In the simulation, the mechanism that preserves the privacy of the ground truth is 3. Whenever a new query is fired from an adversary, the data curator returns the

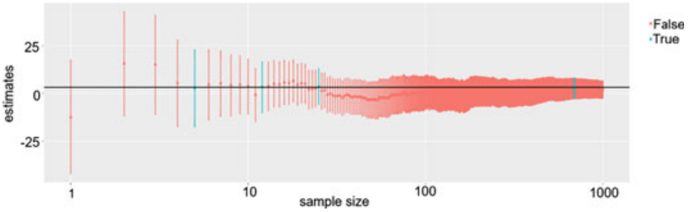


Fig. 3 Confidence intervals of averages of an $\varepsilon = 0.05$—differential privacy mechanism over 1000 queries with a 95% confidence level [31]

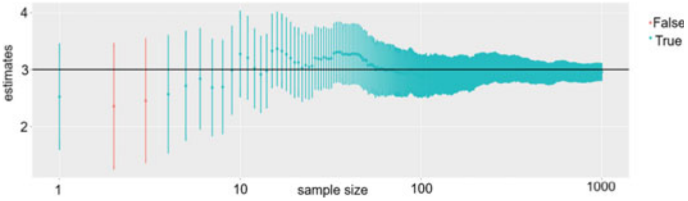


Fig. 4 Confidence intervals of averages of an $\varepsilon = 1$—differential privacy mechanism over 1000 queries with a 95% confidence level [31]

original value or ground truth with additional noise. The noise is drawn from a zero-centered Laplace distribution with a parameter equal to (1/epsilon). In Figs. 3 and 4, the x-axis represents, the number of consecutive queries that an adversary sends to the curator and is logarithmically scaled. The y-axis represents the average of the query replies that the curator sends out, along with the 95% confidence interval of those queries.

Table 2 shows the confidence interval for different values of the privacy budget over a thousand queries with a 95% confidence level. We can say that a decrease in privacy budget (ε) leads to a decrease in accuracy. Considering 0—differential privacy that has low accuracy but protects the privacy very well. In addition, 1—differential privacy doesn't protect privacy but has high accuracy, making both the models useless.

If we consider the value of $\varepsilon = 0$, then it is equivalent to $(1 + e)$. Now, we can write Eq. 1 as Eq. 3. We can say that for $\varepsilon = 0$, both probabilities are the same that is a zero probability loss.

$$Pr[M(x) \epsilon S] \;\leq\; (1 + e)\; Pr[M(y) \epsilon S] \tag{3}$$

5 Conclusions and Future Research Directions

In this article, we discuss the need for differential privacy mechanisms. We formally present the notion of privacy and discuss the state-of-the-art techniques available to ensure the privacy of data collected for various kinds of analysis. Although there are many techniques that can provide the privacy of data, none of them are foolproof. One of the ways to ensure the privacy of data is differential privacy. In addition, we provide the state-of-the-art differential privacy mechanisms, and we discuss different applications of differential privacy. We present the comprehensive survey in the domain of differential privacy. We analyze the state-of-the-art differential privacy mechanisms and present their pros and cons. We provide the challenging research issues related to differential privacy and highlight the research gaps related to differential privacy. We conclude the article by emphasizing the future research directions in the area of differential privacy such as the use of cryptography to facilitate privacy-preserving data analysis, the novel and efficient design of differential privacy algorithms, and the design of adjustable and configurable privacy budgets.

References

1. Apple's 'differential privacy' is about collecting your data—but not your data, https://www.wired.com/2016/06/apples-differential-privacy-collecting-data/
2. Andrés ME, Bordenabe NE, Chatzikokolakis K, Palamidessi C (2013) Geo-indistinguishability: differential privacy for location-based systems. In: Proceedings of

the SIGSAC conference on Computer & communications security. ACM, Berlin, pp 901–914. https://doi.org/10.1145/2508859.2516735

3. Asoodeh S, Liao J, Calmon FP, Kosut O, Sankar L (2021) Three variants of differential privacy: lossless conversion and applications. J Sel Areas Inf Theory 2(1):208–222. https://doi.org/10.1109/JSAIT.2021.3054692
4. Box D, Hejlsberg A (2007) LINQ: NET language-integrated query. MSDN Developer Centre 89:1–27
5. Clifton C, Anandan B (2013) Challenges and opportunities for security with differential privacy. In: Proceedings of the ICISS: 9th international conference on information systems security, vol 8303. Springer, Kolkata, pp 1–13. https://doi.org/10.1007/978-3-642-45204-8_1
6. Clifton C, Tassa T (2013) On syntactic anonymity and differential privacy. In: Proceedings of the ICDEW: 29th international conference on data engineering workshops. Brisbane, pp 88–93. https://doi.org/10.1109/ICDEW.2013.6547433
7. Corporation M Differential privacy for everyone. Last accessed 29 Oct 2021 https://download.microsoft.com/download/D/1/F/D1F0DFF5-8BA9-4BDF-8924-7816932F6825/Differential_Privacy_for_Everyone.pdf
8. Daemen J, Rijmen V (1999) AES proposal: Rijndael, Last accessed 29 Oct 2021 https://www.cs.miami.edu/home/burt/learning/Csc688.012/rijndael/rijndael_doc_V2.pdf
9. Dalenius T (1977) Towards a methodology for statistical disclosure control. Statistik Tidskrift 15:429–444 March
10. Denning DE (1980) Secure statistical databases with random sample queries. Trans Database Syst (TODS) 5(3):291–315. https://doi.org/10.1145/320613.320616
11. Diffie W, Hellman M (1976) New directions in cryptography. Trans Inf Theory 22(6):644–654. https://doi.org/10.1109/TIT.1976.1055638
12. Domingo-Ferrer J (2008) A survey of inference control methods for privacy-preserving data mining. In: Proceedings of the privacy-preserving data mining: models and algorithms. Springer, pp 53–80. https://doi.org/10.1007/978-0-387-70992-5_3
13. Dwork C (2006) Differential privacy. In: Proceedings of the ICALP: international colloquium on automata, languages, and programming, vol 4052. Lecture Notes in Computer Sciences, Springer, Venice, pp 1–12. https://doi.org/10.1007/11787006_1
14. Dwork C (2011) Differential privacy. In: Encyclopedia of cryptography and security, 2nd edn. Springer, Venice, pp 338–340. https://doi.org/10.1007/978-1-4419-5906-5
15. Dwork C, Kenthapadi K, McSherry F, Mironov I, Naor M (2006) Our data, ourselves: privacy via distributed noise generation. In: Proceeding of the EUROCRYPT: advances in cryptology, St. Petersburg, vol 2004. Springer, pp 486–503. Lecture Notes in Computer Science. https://doi.org/10.1007/11761679_29
16. Fioretto F, Van Hentenryck P, Zhu K (2021) Differential privacy of hierarchical census data: an optimization approach. Artif Intel 296:1–20. https://doi.org/10.1016/j.artint.2021.103475
17. Gohari P, Wu B, Hawkins C, Hale M, Topcu U (2021) Differential privacy on the unit simplex via the Dirichlet mechanism. Trans Inf Forensics Sec 16:2326–2340. https://doi.org/10.1109/TIFS.2021.3052356
18. Goldwasser S, Micali S (1984) Probabilistic encryption. J Comput Syst Sci 28(2):270–299. https://doi.org/10.1016/0022-0000(84)90070-9
19. Hu X, Yuan M, Yao J, Deng Y, Chen L, Yang Q, Guan H, Zeng J (2015) Differential privacy in telco big data platform. VLDB Endowment 8(12):1692–1703. https://doi.org/10.14778/2824032.2824067
20. Jain P, Gyanchandani M, Khare N (2016) Big data privacy: a technological perspective and review. J Big Data 3(1):1–25. https://doi.org/10.1186/s40537-016-0059-y
21. Jain P, Gyanchandani M, Khare N (2018) Differential privacy: its technological prescriptive using big data. J Big Data 5(1):1–24. https://doi.org/10.1186/s40537-018-0124-9
22. Lee DGY (2008) Protecting patient data confidentiality using differential privacy. Last accessed on 30 Nov 2021 https://scholararchive.ohsu.edu/concern/etds/2f75r8056
23. Lemmens A, Croux C (2006) Bagging and boosting classification trees to predict churn, vol 43. SAGE Publications, pp 276–286. https://doi.org/10.1509/jmkr.43.2.276

24. Li H, Dai Y, Lin X (2015) Efficient e-health data release with consistency guarantee under differential privacy. In: Proceedings of the 17th international conference on e-health networking, application & services (HealthCom). IEEE, Boston, pp 602–608. https://doi.org/10.1109/HealthCom.2015.7454576
25. Majeed A, Lee S (2021) Anonymization techniques for privacy preserving data publishing: a comprehensive survey. IEEE Access 9:8512–8545. https://doi.org/10.1109/ACCESS.2020.3045700
26. Mallya PVS, Ajith A, Sangeetha T, Krishnan A, Narayanan G (2020) Implementation of differential privacy using diffie–hellman and AES algorithm. In: Proceedings of ICICCT: inventive communication and computational technologies, Hyderabad, vol 89. Lecture Notes in Networks and Systems, pp 143–152. https://doi.org/10.1007/978-981-15-0146-3_15
27. McSherry FD (2009) Privacy integrated queries: an extensible platform for privacy-preserving data analysis. In: Proceedings of the international conference on management of data. SIGMOD '09, ACM, Rhode Island, pp 19–30
28. Merkle RC (1978) Secure communications over insecure channels. Commun ACM 21(4):294–299. https://doi.org/10.1145/359460.359473
29. Mohan P, Thakurta A, Shi E, Song D, Culler D (2012) GUPT: privacy preserving data analysis made easy. In: Proceedings of the ACM SIGMOD international conference on management of Data. ACM, SIGMOD, Scottsdale, pp 349–360. https://doi.org/10.1145/2213836.2213876
30. Narayanan A, Shmatikov V (2008) Robust de-anonymization of large sparse datasets. In: Symposium on security and privacy. IEEE, Oakland, pp 111–125. https://doi.org/10.1109/SP.2008.33
31. Nguyen A Understanding differential privacy, https://towardsdatascience.com/understanding-differential-privacy-85ce191e198a, Last accessed 29 Oct 2021
32. Roy I, Setty ST, Kilzer A, Shmatikov V, Witchel E (2010) Airavat: security and privacy for mapreduce. In: NSDI, vol 10, pp 297–312
33. Sarathy R, Muralidhar K (2011) Evaluating Laplace noise addition to satisfy differential privacy for numeric data. Trans Data Privacy 4(1):1–17. https://doi.org/10.5555/2019312.2019313
34. Xu J, Zhang W, Wang F (2021) $A(DP)^2SGD$: asynchronous decentralized parallel stochastic gradient descent with differential privacy. Trans Pattern Anal Mach Intel. https://doi.org/10.1109/TPAMI.2021.3107796

Fitness Freaks: A System for Detecting Definite Body Posture Using OpenPose Estimation

Harshwardhan Pardeshi, Aishwarya Ghaiwat, Ankeet Thongire, Kiran Gawande, and Meghana Naik

Abstract Nowadays, because of busy schedules, people have no time to go to the gym, and even if they manage to find a gym nearby having a gym trainer besides all the time to correct postures while doing exercise is impossible unless people do not opt for a personal trainer. Even if they assist a personal trainer for them, they would have to adjust their time accordingly, and both of these methods are quite costly, and not everyone can afford it; also, during this global pandemic, people are stuck at home and have no access to go to the gym or can't even take a risk of getting in contact with a personal trainer. Performing exercises let it be exercise or doing Yoga proper body posture is important; if not performed properly, it can lead to crucial problems such as poor joint alignment, increased shear forces on the spine, compression of disks and joints, less space for nerves to course through the body due to compression, and reduced blood flow to prevent such injuries and pains, and to track gym exercises repetitions, we came up with a system called Fitness Freaks. Fitness Freaks is an AI fitness tracker. It tracks users' body movements using human pose estimation. This in turn keeps track of repetitions of gym exercises and detection of wrong body posture while doing Yoga.

Keywords Computer vision · Fitness · Gym exercises · OpenPose · Posture · Yoga

H. Pardeshi (✉) · A. Ghaiwat · A. Thongire · K. Gawande · M. Naik
Computer Engineering Department, Sardar Patel Institute of Technology, Andheri (West), Mumbai 400059, India
e-mail: harshwardhan.pardeshi@spit.ac.in

A. Ghaiwat
e-mail: aishwarya.ghaiwat@spit.ac.in

A. Thongire
e-mail: ankeet.thongire@spit.ac.in

K. Gawande
e-mail: kiran_gawande@spit.ac.in

M. Naik
e-mail: meghana_naik@spit.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*, Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_76

1 Introduction

Development in the field of fitness and well-being of health has grown exponentially in the last decade which includes fit-bands, calorie counter, diet planner, and run tracker. Supporting the advancement in this field, we focused on the problem of getting assistance while doing exercises and focusing on the prevention of injuries. A system to track count of user performed exercises and detect errors in a Yoga pose, with the use of computer vision in which we use OpenCV Python library, OpenPose which uses baseline CNN network, and COCO dataset. We defined rules for different Yoga poses, exercises, and accordingly, we get the results for the count of repetitions of the user-performed exercises and detect error for the Yoga poses; if there is any, feedback is given to users based on the results observed. For this, the user needs to run the system on their laptop, no fitness band or other extra weight is to be carried with the user while performing it. After analyzing the user's exercise, feedback about the exercise is given to the user using a voice assistant. It helps the user to keep track of repetitions and focus more on correcting the body posture while doing exercises.

2 Literature Survey

Xiong et al. [1] in 2020 worked on robust vision-based workout analysis. Test results show the prevalence of their proposed 3D posture assessment over the past ones. It identifies incorrect motions but not along with the timing of these motions and hence does not provide timely feedback to users. They could not integrate their model with video tutorials.

Yadav et al. [2] in 2019 approached to accurately recognize various Yoga poses using deep learning algorithms A dataset of 6 Yoga asanas had been created using 15 individuals. A hybrid deep learning model was proposed using CNN and LSTM for Yoga recognition on real-time videos; the system can be implemented on a portable device for real-time predictions and self-training.

Gu et al. [3] in 2019 adopted deep learning models for human pose estimation and worked on home-based physical therapy with an interactive computer vision system. They could not provide users a side-view option and could not develop an algorithm that gives more detailed feedback on how the patient is doing, i.e., instead of giving feedback based on the overall performance.

Chen et al. [4] in 2018 incorporated computer vision strategies and proposed system that examines the practitioner's stance from both front and side perspectives by separating the body shape, skeleton, dominant axes, and points. Improving or even redesigning the methods of feature point detection and assistant axis generation for some poses can make the system more solid.

Chen and Yang [5] created this application for correcting users' pose by creating ideal movements of exercise. They used deep neural networks and OpenPose for pose estimation and machine learning and heuristic-based models for getting the result of performance by comparison. The application created works only on the Web which runs on Windows and Linux computers based on GPU.

In this paper, Keshari [6] has used opencv for image processing, SVM and RCNN for detecting the errors. They have created their own dataset for detecting the errors; they distributed the dataset to understand the proper posture and remaining for testing using SVM and RCNN to detect wrong posture.

In this paper, Nagarkoti et al. [7] use a pre-recorded trainer's video using deep learning and opencv. For tracking users body movements, optical flow tracking is used; dynamic time warping is used to sync trainers and users body movements. It only corrects user posture; proper AI assistant for the user should include tracking exercises repetitions, errors, and creating reports.

Agrawal et al. [8] used various classification techniques to detect Yoga poses out of which random forest classifiers gave the best results. It detects different Yoga poses using this application along with identification of the pose; if the accuracy is calculated, then the user will be able to track and improve performance.

In Cao [9] did pose estimation for single person and multi-person. In single person, they perform inference over a combination of local observations on body parts and the spatial dependencies between them. For multi-person, they have used a top-down strategy to first detect people and then estimate the pose of each person independently. It works only on an image and cannot be used on a video.

Kumar et al. in [10] proposed to use OpenPose on the client's ongoing and recorded sessions to distinguish the joint areas by utilizing Part Confidence Maps and Part Affinity Fields. Then, based on the difference in angles, feedback is provided to the user. This works only when images are provided by the user; it does not work in real-time.

In [11] Chiddarwar et al. collect a single ideal image for getting the key points and store these locally in their machine. Then, OpenCV is used to predict the 17 essential keypoints using a pretrained model; the distance between each body part is calculated, using Euclidean distance. The system only detects the Yoga pose and gives the correctness of the pose.

In [12] Dang et al. did a survey, on human pose estimation methods. The single-person estimation is classified into two types: regression-based approach and heatmap-based approach. Multi-person estimation is classified into two categories: top-down approach and bottom-up approach. Now-a-days, the human pose estimation methods are improved significantly and can still be improved to use them for real-world applications. The speed of algorithms is still slow for real-time prediction.

In [13] Sajjad et al. compared different techniques for human pose recognition to identify which is better by calculating accuracy for each and getting better results so that the implementation of the human pose recognition should be correct.

3 Proposed Methodology

After identifying the problem statement and the gaps from the research paper, we proposed a solution to build a tech-based personal trainer which will guide and track users' exercises 24/7.

Fitness Tracker: The fitness tracker will help maintain users' body posture while doing gym exercises and Yoga poses.

Gym Exercises: The system counts the repetitions of the exercises performed so that the user does not have to keep track of the exercises.

Yoga poses: The system tells the accuracy of Yoga poses performed and detects an error in users' posture.

We got assistance from 12 professional trainers and Yoga instructors who could help us build a robust system. We asked them to identify all the factors that needed to be considered while doing the 2 simple exercises, one a gym exercise and another a Yoga pose. In gym exercise, we have built a system for bicep curls, whereas in Yoga pose built a system for Warrior Pose. Based on the guidance, we created 10 rules for Warrior Pose as the total number of factors involved for performing that pose were 10 and similarly, for bicep curls, only 2 rules; based on those 2 rules, the user's repetitions are going to be counted. If the user does not satisfy those rules, the repetitions won't be counted.

3.1 Requirements

The system is a Web site-based application that runs on a laptop. The user needs enough space to perform the exercise and place the camera so that the whole body of the user is fitted in the camera angle.

3.2 Assembling Data

For both gym and Yoga poses, we have taken a total of more than 150 images of professional trainers and Yoga instructors to build our system that can identify ideal values of the factors involved in the exercises. OpenCV is a Python library. It is used in the system to focus on capturing the video and detecting the face. It is also used for image processing. OpenCV uses the COCO dataset which is the largest dataset for 2D pose estimation. It has around 1.5 million object instances, 80 categories of images consisting of objects, and around 250,000 instances of people. The COCO dataset is also considered a large-scale universal dataset for a lot of tasks related to computer vision.

3.3 Data Preprocessing

In this computer vision library, OpenCV is used where we use OpenPose for human pose estimation. Human pose estimation tracks and recognizes different main points of posture in different individuals. We use the videos of instructors performing the exercises. First, it analyzes the picture from the user's video, and it acts as an input image. Initially, to extract feature maps of the input, the picture is passed through baseline CNN. It utilizes the VGG-19 network's first 10 layers. To produce the Part Confidence Maps and Part Affinity Field, the feature map is then processed in multistage CNN.

3.4 Algorithm

A greedy bipartite matching algorithm is used to process the Confidence Maps and Part Affinity Fields that are created above and are used to get the postures for every individual in the picture.
Certainty Maps: A Confidence Map is a 2D portrayal of the conviction that a specific body part can be situated in some random pixel.
Part Affinity Fields: Part Affinity is a bunch of 2D vector handles that encode the location and direction of body portions of various individuals in the picture. It encodes information as pairwise associations between body parts.
Multi-Stage CNN: The above multi-CNN engineering has three significant stages:
The original set of stages anticipated the Part Affinity Fields refines Lt from the element guides of base organization.
The second arrangement of stages takes utilizes the yield Part Affinity Fields from the past layers to refine the forecast of certainty maps identification.
The last certainty guides and Part Affinity Field are then passed into the covetous calculation for an additional cycle.

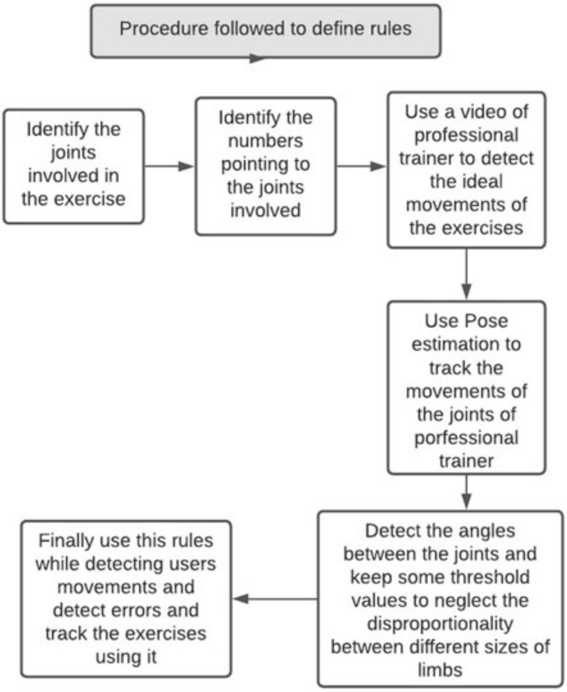
4 Implementation

The system first takes users video as an input; the user is instructed with a demo video showing how to perform the exercise; user can watch the video and perform the exercise, and from that, we find the 10 stillest points and take the mean of those points using human pose estimation.

More than 150 images are used in training our system; furthermore, 30 user's videos of performing the exercise are taken as testing data for the system.

Different rules are being written for exercise and Yoga to define the ideal position of that specific pose using OpenPose. A specific procedure is followed while defining the rules in Fig. 1. First of all, the joints involved in the exercise are identified. Then,

Fig. 1 Procedure followed to define rules


Procedure followed to define rules
Identify the joints involved in the exercise
Identify the numbers pointing to the joints involved
Use a video of professional trainer to detect the ideal movements of the exercises
Use Pose estimation to track the movements of the joints of porfessional trainer
Detect the angles between the joints and keep some threshold values to neglect the disproportionality between different sizes of limbs
Finally use this rules while detecting users movements and detect errors and track the exercises using it

the number pointing to the joints is identified with the help of the COCO human pose estimation model. Then, a video of a professional trainer is used to detect the ideal movements of the exercise and Yoga. Further, pose estimation is used to track the movements of the joints of the professional trainer. The angles between the joints are detected, and some threshold values are kept to neglect the disproportionality caused due to various body types and size depending upon the gender and age.

In Fig. 2 there is exercise as well as Yoga detection under one roof. For bicep curl exercise, required distance, angle, and movement of limbs are taught to the system using trainer's videos, and for warrior II Yoga pose, professional Yoga instructor pose is recorded inside the system. Once the rules are written, (Fig. 3) captured video of the user is passed into the system, and the moments are detected using computer vision, i.e., pose estimation by the means of OpenCV. Rules are being checked based on the exercise performed using TensorFlow. Errors are detected, and repetitions are tracked for the bicep curl exercise. For the Warrior II Yoga pose, errors are detected in real time, i.e., pointed out and real-time feedback is given by a voice assistant to correct the Yoga pose. A feedback report is generated based on the user performance. In this way, using the defined rules, user movements and errors are detected, and track of the exercise and Yoga is kept.

We use OpenPose in our project and write rules needed for calculating the angles, slopes, min, and max ratios. If the conditions are not matching, i.e., if the user is not performing Yoga properly then feedback is given to them on the basis of rules

Fig. 2 Flowchart of proposed system

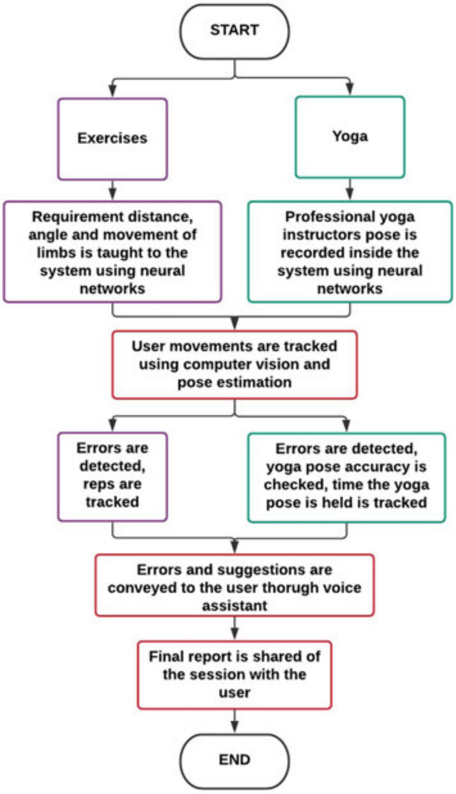
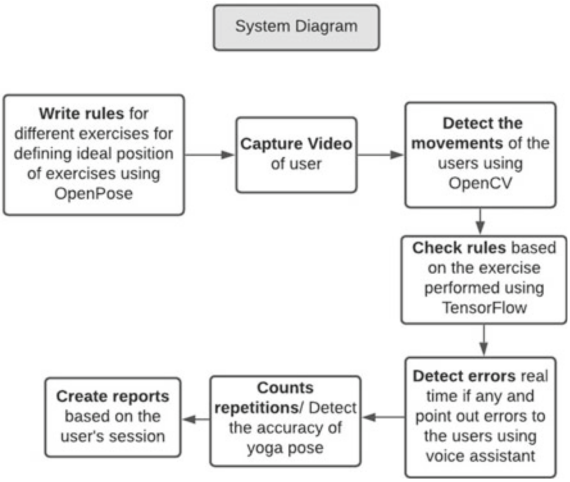

START
Exercises
Yoga
Requirement distance, angle and movement of limbs is taught to the system using neural networks
Professional yoga instructors pose is recorded inside the system using neural networks
User movements are tracked using computer vision and pose estimation
Errors are detected, reps are tracked
Errors are detected, yoga pose accuracy is checked, time the yoga pose is held is tracked
Errors and suggestions are conveyed to the user thorugh voice assistant
Final report is shared of the session with the user
END

Fig. 3 System diagram


System Diagram
Write rules for different exercises for defining ideal position of exercises using OpenPose
Capture Video of user
Detect the movements of the users using OpenCV
Check rules based on the exercise performed using TensorFlow
Detect errors real time if any and point out errors to the users using voice assistant
Counts repetitions/ Detect the accuracy of yoga pose
Create reports based on the user's session

defined. Count of exercise is tracked if conditions are satisfied. The system will provide instructions on how to perform the pose. Once the user is ready, he/she records a video of themselves doing the pose (using a Webcam). Using OpenPose, the video input is processed and an open source; deep learning-based library is used for key-point detection. The app uses the key points obtained from the video to classify different errors that occurred while doing the pose with the help of a rule-based system. Repetitions of exercise are tracked, and feedback is provided to the user so that they can safely improve their pose.

To check the accuracy, we have established 10 rules. Table 1 describes the evaluation factors considered for accuracy calculation. We derived a formula to get the final accuracy. For 10 rules, we have 10 variables for users' pose and 10 for the correct Yoga posture which we get by examining the pose of professional Yoga instructors. We derive the formula as

$$\text{Error} = \frac{|\text{Users' value} - \text{Ideal Value}|}{\max(\text{User's value, Ideal Value})} \tag{1}$$

The individual errors of each rule are recorded using the above formula; we assume the weight for each rule is the same for detecting error, and the mean of all the errors is taken to denote the final error, and accordingly, we get the accuracy of the pose. The 10 rules together check if the warrior pose performed by the user is correct or not, based on which we derived a formula for checking accuracy.

$$\text{Error} = \frac{\sum_{i=1}^{10} \frac{|x_{ui} - x_{pi}|}{\max(x_{ui}, x_{pi})}}{10} \tag{2}$$

where, i, denotes the rule number among the 10 rules for warrior pose
x_{ui}, denotes value of returned by the functions of ith rule for users pose
x_{pi}, denotes value of returned by the functions of ith rule for professional Yoga instructors pose.

Table 1 Evaluation factors considered for accuracy calculation

Rule number	Evaluation factors considered
1	Slope of the arms
2	Angle between the hip, knee, and ankle
3	Angle between the hip, knee, and ankle
4	Ratio of distance between eyes to distance between ears
5	Slope of left hip and right hip
6	Ratio of arm distance to legs distance
7	Slope between left shoulder to neck and neck to right shoulder
8	Slope of neck and mid-hip
9	Slope of neck and mid-hip
10	Ratio of arm distance to legs distance

Table 2 Evaluation factors considered for accuracy calculation of exercise: Bicep curl

Rule number	Evaluation factors considered
1	Angle between wrist, elbow, and shoulder

The rules are:

1. Keeping the arms straight and palms faced down
2. Keeping front leg straight perpendicular to the floor
3. Ensuring the front knee doesn't extend beyond the ankle and is inline with the heel
4. Keeping face inline with front hand
5. Distance between the feet is wide enough, so the legs get stretched
6. Keeping shoulders down, stretched out and not lifted toward ears
7. Keep hips and shoulders faced sideways toward camera
8. Keep hips and shoulders in same line so that the rib cage isn't floating forward
9. Keep hips and shoulders in same line so that the rib cage isn't floating backward
10. Place back leg straight and strong and keep short distance between the legs.

Table 2 depicts the evaluation factors considered for the accurate calculation of bicep curl. For bicep curl detection, we have set a threshold range of angle between wrist, elbow, and shoulder depending upon the videos of the trainer and a counter for counting the number of repetitions of the exercise performed.

5 Results

To explain the results of the system built, we take an example of some users performing Warrior Pose II. The values of different factors which are mentioned in Table 1 for the users performance are shown in Table 3. From the user's value and ideal value, we calculate the error and accuracy of the user's Yoga pose. Each rule has functions written to measure the values such as the slope of arms, angle between hip, knee, and ankle as we get coordinates of all the joints from which we can define functions and calculate different angles, slopes, and ratios to built the system.

$$\text{Error} = \frac{1.3466}{10} = 0.13466 \tag{3}$$

$$\text{Accuracy} = 100 - \text{Error} * 100 = 86.534\% \tag{4}$$

The accuracy of the example taken of user's pose is 86.534%.

Table 3 Accuracy calculation of Yoga: Warrior Pose II

Rule number	Evaluation factors considered	User value	Ideal value
1	Slope of the arms	0.267	0
2	Angle between the hip, knee and ankle	101	90
3	Angle between the hip, knee and ankle	101	90
4	Ratio of distance between eyes to distance between ears	0.67	0.5
5	Slope of left hip and right hip	−0.12	0
6	Ratio of arm distance to legs distance	0.692	0.75
7	Slope between left shoulder to neck and neck to right shoulder	6.52	4.25
8	Reverse slope of neck and mid-hip	0.012	0
9	Reverse slope of neck and mid-hip	0.012	0
10	Ratio of arm distance to legs distance	0.692	0.75

Table 4 Accuracy calculation of exercise: bicep curls

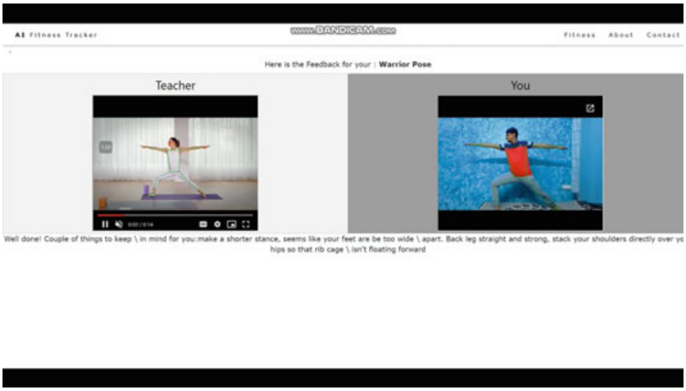
Rule number	Evaluation factors considered	User value	Ideal value
1	Angle between wrist, elbow, and shoulder	216 and 289	Less than 230 and greater than 280

For Warrior II Yoga pose, we have set 10 rules and some ideal values and based on that system provides feedback to the user if the user performs Yoga properly, no correction will be shown to user, and user can continue performing the Yoga without having any concern about getting muscle strain, joint pain, or any sort of injuries which is the main problem our system focuses on. There may be no errors in the user's posture, but the accuracy is not 100% because accuracy is calculated by comparing the user's pose with the ideal pose. Accuracy will increase when the user repetitively practices Yoga and increase flexibility.

For bicep curl detection, Table 4 shows the ideal values set for a threshold range of angle between wrist, elbow, and shoulder depending upon the videos of the trainer and a counter for counting the number of repetitions of the exercise performed.

Figure 4 shows the Yoga: Warrior II Pose. On the left side of the system, the video of the professional trainer is shown so that the user can refer to it while performing the Yoga pose. On the right side of the system, the user pose evaluation is done and according to the proposed rules and feedback is given.

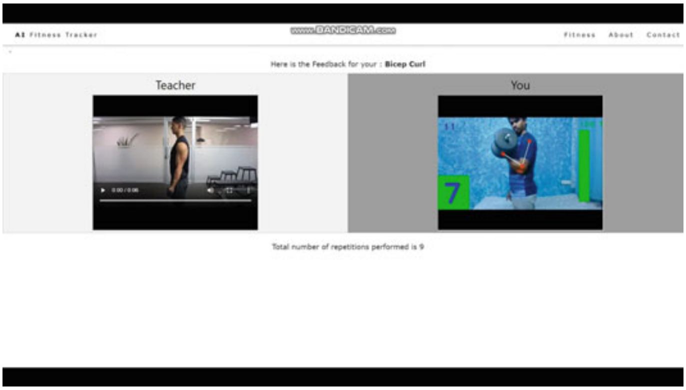
Figure 5 shows the exercise: bicep curl. On the left side of the system, the video of the professional trainer is shown so that the user can refer to it while performing the Yoga pose. On the right side of the system, the user pose evaluation is done and according to the proposed rules and no of reparations are counted.



Here is the Feedback for your : **Warrior Pose**

Well done! Couple of things to keep \ in mind for you:make a shorter stance, seems like your feet are be too wide \ apart. Back leg straight and strong, stack your shoulders directly over yo hips so that rib cage \ isn't floating forward

Fig. 4 Yoga: warrior II



Here is the Feedback for your : **Bicep Curl**

Total number of repetitions performed is 9

Fig. 5 Exercise: bicep curl

6 Conclusion

The system will combine fitness and technology to successfully bring advancement in tracking and acting as a personal trainer to the users without actually requiring the help of any actual trainer. It brings full-fledged flexibility to the user to perform exercise anytime throughout the day as per their availability of free time. The constant need for the attention of personal trainers in gyms and Yoga classes while performing the exercise is drawn out by AI fitness tracker. Constant monitoring of users' body movement and joints helps correct users' body posture which is one of the most fundamental parts while doing exercises. The existing methodology primarily focuses on the time for which user exercises and not on the user's correct posture, instead focuses on the time for which exercise is performed. The system has found promising results after testing on more than 50 different users. The ideal values are calculated by taking the mean of values of slopes, ratio, and angles from a dataset of more than 100 trainers performing Warrior Pose II.

7 Future Scope

A system like these can replace personal trainers for correcting body posture and the need for their constant attention while performing exercises or Yoga poses. The system can be featured by adding more exercises and Yoga poses to the system. We can create a user portal that keeps a track and record of the data of their previous exercises and Yoga performed in the system's database. Variation can be brought by adding sections for men, women, and children depending on their age group and segregating different sections based on the difficulties of the exercises. The system with all these features can be added as a daily routine to prevent injuries and muscle strains while performing exercises also track the counts of all exercises, and keep a report of the users' performance.

References

1. Xiong H, Berkovsky S, Sharan RV, Liu S, Coiera E (2020) Robust vision-based workout analysis using diversified deep latent variable model. In: 2020 42nd annual international conference of the IEEE engineering in medicine biology society (EMBC), pp 2155–2158
2. Yadav S, Singh A, Gupta A, Raheja J (2019) Real-time yoga recognition using deep learning. Neural computing and applications, vol 31, https://link.springer.com/article/10.1007/s00521-019
3. Gu Y, Pandit S, Saraee E, Nordahl T, Ellis T, Betke M (2019) Home-based physical therapy with an interactive computer vision system. In: 2019 IEEE/CVF international conference on computer vision workshop (ICCVW), pp 2619–2628
4. Huang C, He Y-Z, Hsu C-C (2018) Computer-assisted yoga training system. Multimed Tools Appl 77:09
5. Chen S, Yang R (2018) Pose trainer: correcting exercise posture using pose estimation
6. Keshari P (2020) Wrong posture detection using opencv and support vector machine
7. Nagarkoti A, Teotia R, Mahale AK, Das PK (2019) Realtime indoor workout analysis using machine learning amp; computer vision. In: 2019 41st annual international conference of the IEEE engineering in medicine and biology society (EMBC), pp 1440–1443
8. Agrawal Y, Shah Y, Sharma A (2020) Implementation of machine learning technique for identification of yoga poses. In: 2020 IEEE 9th international conference on communication systems and network technologies (CSNT), pp 40–43
9. Cao Z, Hidalgo G, Simon T, Wei S, Sheikh Y (2021) OpenPose: realtime multi-person 2d pose estimation using part affinity fields. IEEE Trans Pattern Anal Mach Intel 43:172–186 Jan
10. Kumar D, Sinha A (2020) Yoga pose detection and classification using deep learning. Int J Sci Res Comput Sci Eng Inf Technol, 11
11. Chiddarwar G, Ranjane A, Chindhe M, Deodhar R, Gangamwar P (2020) Ai-based yoga pose estimation for android application. Int J Innov Sci Res Technol 5:1070–1073
12. Dang Q, Yin J, Wang B, Zheng W (2019) Deep learning based 2d human pose estimation: a survey. Tsinghua Sci Technol 24(6):663–676
13. Sajjad F, Ahmed AF, Ahmed MA (2017) A study on the learning based human pose recognition. In: 2017 9th IEEE-GCC conference and exhibition (GCCCE), pp 1–8

Cyber-Awareness for Dummies

Aashka Raval, Avadh Patel, Wandji Christian, and Nishant Doshi

Abstract "Great Powers comes with Great Responsibility"—this line from Spider-Man suits better today while using Internet responsibly. When, during pandemic, people were caged inside their own houses and apartments, the only means of communication was Internet (social media), but it is also a hive for the crimes and criminals who are mentally evil rather than physically. Cyber-frauds have increased a lot after lockdown, and it is raging high cyberattacks and security issues. In this paper, we try to point out some of those attacks.

Keywords Cybersecurity · Privacy · Attackers · Cloud · Virtual private network (VPN) · Malware · Trojan horse · Social engineering · Phishing · Cybercrime

1 Introduction

The fundamental purpose of cybersecurity for dummies is that in today's day-to-day life, the number of gadgets we use with Internet connectivity has increased dramatically. From the age of three to ninety, everyone owns at least one such gadget that is connected to the Internet and contains sensitive information about the user or their family. In this COVID circumstance, we have noticed a massive shift towards NET Banking, more virtual socialization, online learning (E-Learning), and even access to identity-related documents or data (e-Aadhaar Card) online.

A. Raval (✉) · A. Patel · W. Christian · N. Doshi
Department of Computer Science and Engineering, Pandit Deendayal Energy University, Gandhinagar, Gujarat 382421, India
e-mail: aashka.rmtcs20@sot.pdpu.ac.in

A. Patel
e-mail: avadh.pmtcs20@sot.pdpu.ac.in

W. Christian
e-mail: wandji.cmtcs20@sot.pdpu.ac.in

N. Doshi
e-mail: nishant.doshi@sot.pdpu.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_77

Now, dummies are those who lack understanding of computers and the dangers associated with their usage, since they are unaware of the many cybercrimes that exist as a result of the use of such Internet-connected gadgets, such as identity theft, financial frauds, and so on [1].

Cyberspace provides many options for information and communication, entertainment, education, exchange, and commerce. However, users of the Internet superhighway are subject to a variety of threats that may compromise the security of people, corporations, and the state. Cyber-terrorism, online fraud, espionage, scamming, intrusion, phishing, spamming, viruses, malware, and piracy were recognized as some of the dangers inherent in Cameroon's cyberspace by the legal, political, and strategic challenges. Additional international threats include stolen or leaked intellectual property, stolen funds, stolen computer resources, stolen business information, account data breaches, employee information, email dumps, DDoS attacks, cyberbombs, back doors, marketplace fraud, physical theft/sabotage, audio surveillance, brand, and phishing [2].

1.1 Recent Example

WhatsApp accounts used to get messages from recognized IDs, and when we opened those messages, a Trojan was installed into our system, compromising our data.

Instagram hacking: Here, they send you a DM informing you that you have violated an Instagram guideline and requesting that you re-login. This is a false page established by hackers that collects your login information and hacks your account. This made headlines when YouTube ROST ER Carminati's account was hacked, since he gained notoriety for amassing the most views on certain videos [3].

Now that we have established that both platforms were used by ordinary people who are unaware of such attacks and how to stay vigilant against them, our objective is to conduct research on the most prevalent attacks, classifying them according to the people against whom they are most frequently conducted, and identifying some techniques or steps that can assist them in remaining safe and vigilant against such attacks by following simple Internet safety precautions or being more aware of cyberattacks. The methods should be basic enough that a guy selling groceries in India, as well as a business owner, can grasp them.

2 Literature Survey

- The adverse effect of watering hole attack in distributed systems and the preventive measures.
- IT and cybersecurity awareness—raising campaigns.
- Spyware Guide, http://Spywareguide.Com, Accessed 2015–02–10.

- International Journal of the Institution of Engineering and Technology (IET) On Information Security, Vol. 8, Pp. 18–24, England, 2013.
- IEEE International Conference on Information Security South Africa (ISSA), Pp. 1–8, South Africa, 2011.
- The study discusses misleading phishing attacks. The most prevalent sort of phishing fraud is deceptive phishing. These scams take place when a reputable source contact you with the intent of compromising your information. Typically, these emails will ask you to: Confirm the account's details. Re-enter data, such as usernames and passwords.
- Phishing attack, watering hole attack, and spyware attack we research on them.
- How to affect to other people and how to survive when attack has been done with us.
- Who is the target of these attacks?
- How to spread awareness regarding these attacks.

3 Work that Has Been Proposed

- Collecting nodal agency reports.
- For the purposes of cybersecurity and government.
- Working to acquire knowledge of various attacks and to understand how they happen.
- Creating dummy attacks.
- Taking steps that assist dummies in remaining aware and avoiding common errors during such attack.
- For instance, take consistent steps to authenticate emails and avoid using any unknown or dubious emails.
- Testing those steps with a group of people who are both acquainted with and unacquainted with the procedure.
- Making report analysis of its work or effect with known people comparing it with unknowns.
- Making a prototype of device that gives training to dummies, how to be aware of cyberattacks with the steps we develop.

4 Procedures/Analysis

4.1 Field of Study

Cybercrime is pervasive in India reported by Government and Private Sector Cybersecurity Agencies. India ranked third in terms of Internet user growth till 2017, according to research by NITI Aayog, indicating that it is relatively simple to target the Indian populace due to a lack of awareness about cybercrime and Internet use.

4.2 Cybercrime is a Big Problem Around the Globe

Cybercrime is prevalent around the globe and mostly targets ordinary people. Three attacks are described below:

- Phishing attack
- Watering hole attack
- Spyware attack.

Phishing:

Throughout this epidemic, we have witnessed an increase in fraud instances, particularly those involving cybercrime and, most likely, phishing. Now, phishing is a relatively simple act in which a criminal sends you a simple email or other means, a fraudulent link that looks very similar to the social media accounts you use, banking websites, or other trustworthy websites and steals information such as your name, password, ID, and contact details in exchange for financial or informational gain [4].

Phishing is a kind of social engineering. Phishing is carried out in a very methodical fashion; the attacker always sends a very legitimate email and includes some bogus link that links you to a false web page [5].

a. The perpetrator sends the victim or victims an email.
b. The victim opens the email, which contains phone links.
c. The link links them to a scam site, where the victim divulges all pertinent information.
d. The attacker now collects all of the victim's information and navigates to the original site in order to get access to and control of the victim's bank account, social media account, and so on (Fig. 1).

Phishing begins with an email or other kind of contact intended to assist in assaulting the target. The communication is sent as though it came from a trustworthy source. If a victim falls for the scam, he or she is send to a spam website

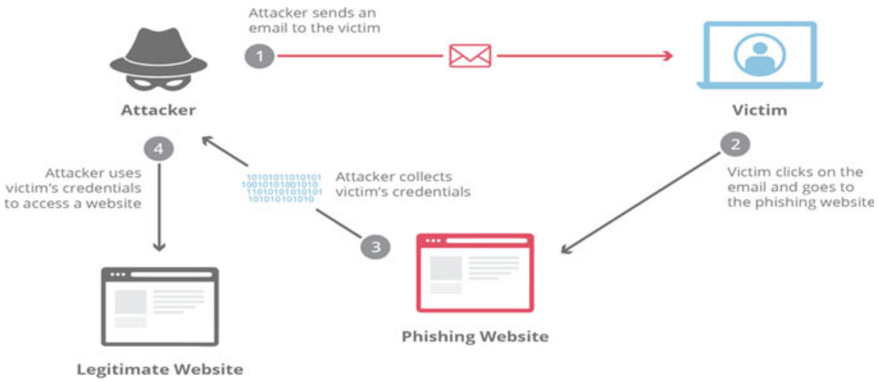

Attacker sends an email to the victim
1
Attacker
Victim
2
Victim clicks on the email and goes to the phishing website
4
Attacker uses victim's credentials to access a website
Attacker collects victim's credentials
3
Phishing Website
Legitimate Website

Fig. 1 Process of phishing

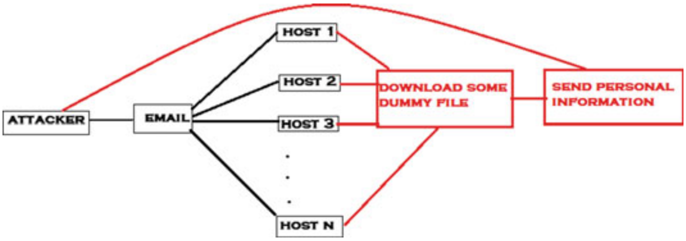


Fig. 2 Email as a result of an assault

with your personal information. Malware is sometimes transferred onto the victim's machine as well [6]. In this email or today, we notice Instagram direct messages or Facebook messages sent to multiple individuals, requesting their bank account information or other personal information using a forged ID or by building phoney sites and stealing their information (Fig. 2).

For instance, a recent article by CERT-in (The Indian Computer Emergency Response Team) warns consumers of the possibility of phishing attempts when they get an email offering "Free Covid-19 Test throughout India".

Types of phishing attack

1. Phishing.

   This is the most common kind of attack; the attacker impersonates a legitimate business and then seeks to steal the victim's data using the company's name. Following their acquisition of the data, they exploit its sensitivity and seek to coerce the victim into acting in their favour [7].
2. Spear phishing.

   By the term "spear phishing", we mean assaults that are as precise and targeted as a spear. In this sort of assault, the victim is a targeted organization or individual whose information may be used to benefit the attacker. To be specific, this attack is not carried out by a random attacker but by someone who knows basic information about you and can profit from your information [7].
3. Clone phishing.

   Clone phishing is a very common kind of phishing assault since it involves the creation of a clone, or cloning, of a previously given valid male for phishing purposes. The only difference is the link to the clone email, which they provide to victims through a similar-looking email account. This technique may be used to pivot away from the infected system and steal all the data, or to establish a foothold on another computer [8].

   A case study of spear phishing:
4. Phishing email solution:

   - It is necessary to understand the difference between http and https, as the "s" in https represents for security and is confirmed to users yes at point.

   - Symbol or logo may indicate a distinction, but this is not always the case, since logos are forged and various editing programmers make it simple to replace or replicate the watermark or logo [9].
   - Previously, improper grammar or spelling was employed, but this is no longer necessary since we may get the exact same message as previously without making any such error.
   - Placing a message in the Gmail spam folder is not required. Gmail spams all bogus emails since its algorithm spams only those that are regularly reported or identified as harmful based on predetermined criteria [10].

5. Whale hunting.

   Whaling is a kind of phishing attack in which the attacker targets the victim's wealth and prominence; the attacker collects the victim's information through a variety of methods, including social media accounts and then assaults the victim. Additionally, these victims of assault are dubbed "Whales" or "Big Phish". Whale phishing is similar to spear phishing [11].
6. Phishing via telephone.

   - Voice phishing is a kind of phone criminal attack that makes use of social engineering and the telephone system to get sensitive personal and financial information in order to undertake financial transactions. It is referred as "vishing" [12].

Phishing attack prevention

Phishing efforts are usually camouflaged as spam or pop-up windows, making detection difficult. Numerous these are covered in further depth in subsequent parts [10].

- Protect against spam: It is kind of message or mail you get which are probably false mails and sent in bulk that have unwanted links which target you either with any spyware or access details.
- Do not click on links in emails from unknown senders, download files, or open attachments: It is always prudent to adequately safeguard sensitive data, such as bank and social media account information, and to open attachments in emails only if you are expecting them and are aware of their contents, even if you are the sender.

How can phishing attacks be detected?

Although the Internet is a massive resource that people may use to accomplish anything, Facebook, Twitter, Gmail, Dropbox, PayPal, eBay, and bank portals all have phishing twins. Phishing is a term that refers to spoof websites that make an effort to seem to be a legitimate website that you are acquainted with and often visit. Numerous phishing detection strategies are described below [4].

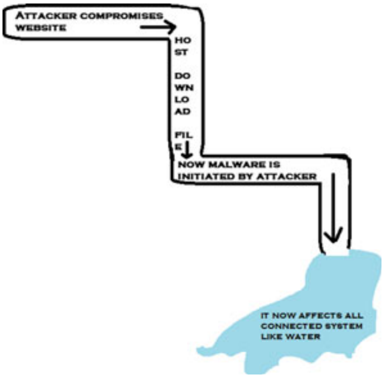
- **Utilize a customized DNS service**: The user may then utilize the DNS resolution service to ensure that they have access to all of the websites they visit. Because your computer is unaware of the location of your Facebook (at least in terms of its Internet Protocol, or IP, address), it must request the IP address from a DNS resolution service. Additionally, there are a number of specialized and independent DNS providers that provide services other than name resolution. They evaluate the site's content and its vulnerability to virus or phishing attacks.
- **Utilize the phishing list provided by your browser**: Modern browsers provide a list of often used phishing sites. The browser uses this list to determine if the website you are now viewing or previously visited is a phishing site. As a result, always check out before browsing any further websites.
- **Use sites to verify links**: Oftentimes, while developing a website or application, a variety of different sorts of connections are employed. If you have provided a link or are dubious about it, you may copy it and verify it on a range of other websites. This may suggest that the site has malicious code or is a phishing victim.
- **Utilize your own Ninja abilities**: This may seem unnecessary, but utilizing your Ninja talents to detect the phishing effort may help you avoid malware or phishing sites that have not been added to your list and would immediately raise a red flag.
- **Investigate safe connections**: This is often indicated by a green area in the address bar, as well as the presence of HTTPS in the URL.

Water holing:

In this attack, they create an identical phoney website that seems similar to the legitimate one and steal the victims' login credentials (Fig. 3).

For instance, "Holy Water" is a water holing assault discovered in the year 2019's previous phase. Even more recently, targeted corporations include Facebook, Apple, and Twitter (in this case, account credentials for 2,50,000 users). Among the most frequent victims of waterholes are the defence industry, the academic sector, government organizations, the financial sector, and utilities [13].

Fig. 3 Assault by water holing


ATTACKER COMPROMISES WEBSITE
HO ST DO WN LO AD FIL E
NOW MALWARE IS INITIATED BY ATTACKER
IT NOW AFFECTS ALL CONNECTED SYSTEM LIKE WATER

How does the watering hole assault work?

Cyberattackers plan their assaults primarily by exploiting the target organization's software security flaws. Those assaults are the best technique to avoid their attacks to update all frequently used software, including the operating system, in an organization. This sort of company should monitor both the incoming and outgoing network traffic; it should also use appropriate security measures. Therefore, stay on the lookout for different cyberattacks in order to defend your system on a better day. Additionally, be safe and secure. The attackers operated on their targets by sending terrible word messages from various members of the community in general or by sending the email "I LUV U". Over time, the attacker targets particular personnel who assist in navigating an organization's personal hierarchy, etc., which results in command and control over the organization's whole infrastructure [13]. The attackers first scoured the websites regularly frequented by a victim or a certain group and then used malware analysis to infiltrate the frequently visited websites. The attackers then discovered the website's weaknesses and inserted malicious programming code, often in the form of JavaScript or HTML design code, into the advertisements, banners, and other content shown on the website. After that, the malicious code forwarded the targeted groups to a phishing site that had malware or malware tenements. When the targeted group accesses certain websites, a malicious script is automatically downloaded to the victim's device. Malware collects victims' personal information and transmits it back to the attacker's command and control server [9].

Who are the WHA'S objectives?

- WHA strategies are being used to target assaults and get sensitive information and intelligence from the entity that is being delisted. The gathered data is then utilized to launch more severe assaults on the targeted companies.
- Leading business organizations [13], non-governmental organizations (NGOs) [13], governing bodies [13], financial institutions [14], and colleges.

Why is this method effective?

In order for the watering hole attacks to be effective, the attackers employ techniques that allow them to bypass certain targeted companies, protection, or firewalls. This might occur as a result of human mistake. Watering hole's primary objective is not to spread malware to the distributed system; rather, the attackers used well-known and reputable websites to direct their intended victims. Watering hole attacks may also include zero-day exploits that target unpatched vulnerabilities, leaving victims with no defence against these exploits [8].

Cybersecurity education

No matter how many systems you put in place to protect against watering hole or spear-phishing assaults, they will be ineffective unless you have a cyber-awareness workforce that is well trained in information security and is capable of recognizing risks and hiding hazards in digital landscapes. Using various tools, such as Keep net's Awareness Educators, you may strengthen your cybersecurity posture by providing enhanced education and cutting-edge information security training [15]. This is coupled with their phishing simulator module to assist your simulator colleagues in engaging with appropriate components and phishing simulations to raise knowledge of sophisticated phishing and other harmful cyberattacks, allowing you to mitigate future risks [16].

How to avoid becoming a victim of a "Watering holing" attack?

- It is advised that you update all of your software to the newest versions and maintain an up-to-date operating system.
- It is recommended that firewalls and other network security solutions be configured appropriately.
- To avoid watering hole attacks, it is important to monitor all popular websites visited by workers and guarantee that they are malware free.
- Ensure that your organization's website is malware free.
- To be secure, use a VPN and your browser's private browsing function to mask your online activity.
- It is always prudent to install security measures to inform users when a website is hacked.
- Additionally, it is important that personnel be educated about watering hole attacks [9].

Adware and spyware

Generally speaking, spyware is a sort of harmful software that is placed on computer systems without the end-user's knowledge or consent. By accessing the computer or device, it obtains very sensitive information and sometimes Internet use statistics, which it then forwards to advertising, external users, or data firms (unauthorized users). The majority of spyware is contentious because, even when deployed for relatively benign purposes, it may infringe on the privacy of an end-user and has the potential to be greatly abused [14].

Malicious software (spyware) can be difficult to detect; frequently, the first indication that a user's computing device is infected with spyware is a noticeable decrease in processor or network connection speed (connectivity), or, in the case of mobile devices, a decrease in data usage and battery life. When this occurs, antispyware applications may provide real-time security by analysing network traffic and blocking malicious data, or they can simply detect and remove spyware that has already been installed on a computer through scans [17].

Spyware classification

Spyware is not a single sort of programme like any other; it is a broad category of harmful software that encompasses a variety of other dangerous applications, including adware, key loggers, Trojans, and mobile information-stealing tools.

- **Trojans**: Trojan horses are malicious software programmes that pose as legitimate applications. Unintentionally, a Trojan victim may install a file masquerading as legitimate software, enabling or providing access to the Trojan to be installed on his/her system. The virus may then delete files, encrypt them for ransom, or provide access to the user's data or piece of information to other parties [15].

- **Keyboard loggers**: Key loggers are a kind of monitoring tool that are often used by hackers and cybercriminals to steal login passwords, personally identifiable information (PII), and sensitive user and enterprise data. It is significant because companies may use it to monitor their employees' computer use; parents can use it to monitor their children's Internet usage; and device owners can use it to monitor any illegal activities on their devices [9].
- **Mobile spyware**: Mobile spyware is very dangerous since it may be sent by text message via the short message service (SMS) or the multimedia messaging service (MMS) and often operates without user interaction.
- **Adware**: Malicious adware is often distributed as part of free software, shareware apps, and utilities that are downloaded from the Internet and installed on a user's device when the user visits an infected website [18].

How spyware works?

Malware will eventually be able to infect any Windows or Mac computer, as well as iOS or Android smartphones. While hackers are reasonably competent at getting into the Windows operating system, they are also growing increasingly skilled at hacking into the Mac iOS operating system. Several of the most common ways to infect a computer or computers are as follows: piracy of pictures, games, and music [14].

In general, spyware takes the form of a software that starts automatically when the device is turned on and continues to operate in the background. The presence of spyware consumes random access memory (RAM) and CPU power and has the potential to generate an infinite number of pop-up adverts, slowing the web browser to a crawl. Destructive software (spyware) now follows Internet browsing history and records words, passwords, and other sensitive and private information such as banking information and credit card numbers in its most malicious form. Finally, this information might be accessed and used to perpetrate identity fraud [14].

Examples of Spyware

The most well-known and often used spyware programmes are as follows:

- Zlob/Zlob Trojan: This sort of malware is very cunning since it uploads itself onto a computer and logs every keystroke the user makes, as well as web searches and history.
- Gator: It is often encountered or is mostly present in file-sharing applications.
- Internet Optimizer: This application, which was more popular during the dial-up period, first promises to increase Internet speed but instead replaces all error and login pages with adverts.

How can spyware be avoided?

We may assert that keeping tight cybersecurity is the most effective method of preventing spyware. Several examples of excellent practises are as follows:

- Ensure that you only download software from reputable sites.
- When installing software, we should carefully read all disclaimers.
- Avoid contact with pop-up advertisements.

4.3 Attacks Analysis

- Phishing, in our opinion, is a kind of social engineering assault that is often used to get user data, such as personal information disguised as login credentials or financial information disguised as credit card numbers. We continually analysed phishing databases, phish tanks, and fraud watch lists.
- It provides a free open API that enables developers and researchers to incorporate anti-phishing data into their apps.
- We analysed the water hole attack and discovered that the malicious HTML file checks for the existence of Internet Explorer 10 with Adobe Flash installed. If Internet Explorer 10 with Adobe Flash installed is detected, the malicious HTLML file loads a malicious flash file.
- Watering hole attacks are often directed against legitimate or famous websites associated with high-profile corporations. In other instances, attackers make their targets genuine websites that are widely viewed.
- Numerous spywares have been created for protection; however, none are capable of offering total security. Thus, it is critical to have antispyware or malware software that protects against all types of spyware now and in future.

5 A Futuristic Perspective

To develop simulated attacks and investigate the local pattern of errors that lead to such assaults. After analysing such assaults and their pattern of errors, develop countermeasures to avert future attacks as well as warning notes that might result in future attacks. For example, we are not to disclose any pins we get with anybody until we have validation from the organization. Now conduct a survey or experiment on dummies to see if the dummies taught with the measures are more susceptible than the dummies with less information. To analyse the output and create a prototype design for a portable device that might assist national security agencies as well as multinational corporations in providing training to their employees so that they are properly prepared to do their jobs, have a countermeasure in place for the majority of assaults.

6 Conclusion

Government-led, non-governmental organizations-led, and school- and organization-led cybersecurity awareness initiatives. Combined with strong security mechanisms, awareness training lays the groundwork for safeguarding your business against more complex hacking techniques such as watering hole attacks. Phishing assault may take a variety of different shapes. The objective of this study is to create detection and prevention approaches that will enable the client to take appropriate action in order to avoid future phishing attempts.

References

1. Champion AC (2016) Information security threats and attacks, pp 1–26
2. Aldawood H, Skinner G (2019) Educating and raising awareness on cyber security social engineering: a literature review. In: Proceedings 2018 IEEE international conference teaching, assessment, learning engineering TALE 2018, no. December, pp 62–68. https://doi.org/10.1109/TALE.2018.8615162
3. Ghasemigol M, Ghaemi-Bafghi A, Takabi H (2016) A comprehensive approach for network attack forecasting. Comput Secur 58:83–105. https://doi.org/10.1016/j.cose.2015.11.005
4. Jakobsson M, Myers S (2006) Phishing and countermeasures: understanding the increasing problem of electronic identity theft, pp 1–699. https://doi.org/10.1002/9780470086100
5. Peltier TR (2006) Social engineering: concepts and solutions. Inf Syst Secur 15(5):13–21. https://doi.org/10.1201/1086.1065898X/46353.15.4.20060901/9542.7.3
6. Mohamed JG, Visumathi J (2020) A predictive model of machine learning against phishing attacks and effective defense mechanisms. Mater Today Proc no. xxxx. https://doi.org/10.1016/j.matpr.2020.09.612
7. Bhavsar V, Kadlak A, Sharma S (2018) Study on phishing attacks. Int J Comput Appl 182(33):27–29. https://doi.org/10.5120/ijca2018918286
8. Khonji M, Iraqi Y, Jones A (2013) Phishing detection: a literature survey. IEEE Commun Surv Tutorials 15(4):2091–2121. https://doi.org/10.1109/SURV.2013.032213.00009
9. Subburaj T, Suthendran K (2018) Digital watering hole attack detection using sequential pattern. J Cyber Secur Mobil 7(1):1–12. https://doi.org/10.13052/jcsm2245-1439.711
10. Bojjagani S, Brabin DRD, Rao PVV (2020) PhishPreventer: a secure authentication protocol for prevention of phishing attacks in mobile environment with formal verification. Procedia Comput Sci 171(2019):1110–1119. https://doi.org/10.1016/j.procs.2020.04.119
11. Bruno L (2019) NITI Aayog. J Chem Inf Model 53(9):1689–1699
12. Ivaturi K, Janczewski L (2011) A taxonomy for social engineering attacks. Int Conf Inf Resour Manag 1(2):1–12
13. Ismail KA, Singh MM, Mustaffa N, Keikhosrokiani P, Zulkefli Z (2017) Security strategies for hindering watering hole cyber crime attack. Procedia Comput Sci 124:656–663. https://doi.org/10.1016/j.procs.2017.12.202
14. Kim T, Yi JH, Seo C (2014) Spyware resistant smartphone user authentication scheme. Int J Distrib Sens Netw 2014. https://doi.org/10.1155/2014/237125
15. Padmavathi G, Divya S (2013) A survey on various security threats and classification of malware attacks, vulnerabilities and detection techniques. Int J Comput Sci Appl 2(04):66–72. [Online]. Available: https://www.semanticscholar.org/paper/A-Survey-on-Various-Security-Threats-and-of-Malware-Padmavathi-Divya/4be826fe79be9986dc6e7a8e0e0cac491a5b9540#extracted

16. He W (2012) A review of social media security risks and mitigation techniques. J Syst Inf Technol 14(2):171–180. https://doi.org/10.1108/13287261211232180
17. Chawla A (2021) Pegasus Spyware—a privacy killer. SSRN Electron J. https://doi.org/10.2139/ssrn.3890657
18. Stewart GK (1977) The human dimension in management. Educ Considerations 5(1). https://doi.org/10.4148/0146-9282.2016
19. Benias N, Markopoulos AP (2017) A review on the readiness level and cyber-security challenges in industry 4.0. South–East Europe design automation, computer engineering, computer networks and social media conference, SEEDA-CECNSM 2017. https://doi.org/10.23919/SEEDA-CECNSM.2017.8088234

Deep Analysis of Attacks and Vulnerabilities of Web Security

Naresh Chillur, Avadh Patel, Sagar Patel, and Debabrata Swain

Abstract The world has become excessively dependent on the Internet. Nowadays, Web security is the primary source of concern in today's competitive business environment. For the global data society, it is considered as the core framework. They are particularly vulnerable to security breaches. Web security refers to the process of defending a Web application layer against attacks by unauthorized users. Using a Web application may lead to a variety of problems, many of which stem from the user's submitting erroneous information. Web security threats have evolved significantly since their inception, and they continue to evolve on a daily basis. In this paper, we examined and evaluated a number of different assaults as well as the vulnerabilities and also detailed issues of Web sites and online Web applications in general.

Keywords Web security · Web security threats · OWASP · Cyber security · Security risks · Top 10 OWASP vulnerabilities

1 Introduction

Web security is a major consideration for Web-based applications. Today, Web security is a legitimate source of concern when it comes to the Internet. As a result, Web applications have a more user-friendly architecture for customers. The search engine on the client computer executes the Web content configuration file.

N. Chillur · A. Patel (✉) · S. Patel · D. Swain
Pandit Deendayal Energy University, Gnadhinagar, Gujarat, India
e-mail: avadh.pmtcs20@sot.pdpu.ac.in

N. Chillur
e-mail: naresh.cmtcs20@sot.pdpu.ac.in

S. Patel
e-mail: sagar.pmtcs20@sot.pdpu.ac.in

D. Swain
e-mail: debabrata.swain@sot.pdpu.ac.in

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_78

The WEB service is required for any information or data to be accessible on the current Internet. Now that the World Wide Web is widely used, the programs and data that are based on Web-based applications are the most frequently targeted by network hackers. According to the relevant reports, the Internet data vulnerabilities are all of the most important ways of ensuring data security dissemination on the Internet [1].

Web site security is critical as an action taken to safeguard and secure Web sites against various types of threats. Due to the owners or managers of the Web site's lack of awareness of the critical nature of Web site security, no effort is made to minimize the risk of threats and vulnerabilities. Penetration testing is a legitimate effort to identify and exploit vulnerabilities in Web sites in order to make them safer [2].

Web services are used to transmit and receive information in our everyday lives by a wide range of applications, from banking apps to government organizations and mobile applications, among many others. Hackers target Web apps because they are the most accessible. Hacked information from organizations that might result in financial loss is the most common strategy used by hackers. They look for weaknesses in network architecture, steal private data and passwords, and hack information from organizations that could result in financial damage. According to the RSA Security Report 2019, fraud on social media platforms via Web apps has surged by 43% [2]. The parts of this paper are organized in such a manner that the first portion explains Web security, the second section provides assaults and some data regarding Web-based attacks, and the last section explains the OWASP flaws.

2 Literature Review

Specifically, the objective of this article was to examine a number of approaches for combating this class of threats and determine why they have not been more successful in the past. It also proposed a more effective method for reducing these types of Web vulnerabilities. It also provides the most effective security methods against the aforementioned assaults [3].

A complete analysis of the application security of the Web was conducted in this article, which included three areas: security threats to Web clients, security threats to Web servers, and data transmission security threats. They also provided a variety of effective and dependable ways for preventing attacks, so ensuring the security of programs and data, and thereby ensuring the safety of Internet-based applications [4].

Web security possible solutions and open issues are summarized in this article, which is divided into appropriate subtitles based on the type of attack that is linked with the issues [5].

In this work, the research suggests and validates a collection of properties obtained from the content and structure of Web sites that may be used as indicators of online security vulnerabilities. The characteristics are used to build a classifier consisting of five machine learning approaches, which is then carried out in this study in order to classify unknown Web apps Canfora and Visaggio [6].

The whole article provides a thorough understanding of Web application security, as well as the vulnerabilities that may be found inside it. Security models that are extensively used in academics, e-commerce, and the present condition of different technologies, such as systems, databases, and phone applications are also discussed in this study [7].

The authors address the issue of selecting appropriate SATs for the purpose of vulnerability detection in online applications. As part of their research, we performed a survey of the most popular predefined analysis tools used by Web application developers [8].

3 What is Web Security?

Web Security: It is a process of preventing unwanted accessibility, destruction, alteration, usage, or interruption of Web sites. One of the terms for online security is “cybersecurity,” which refers to securing a Web site or Web application against cyber dangers, such as malware. As with physical structures like houses, businesses, and government buildings, Web sites and online applications are subjected to security breaches.

3.1 Why Web Security?

The goal of Web security is to protect Web sites, Web servers, and Web applications from being compromised by cyber-criminals. It is the actual process of safeguarding Web sites against the following: unauthorized access, modification, destruction or disruption, and usage.

3.2 Web Security Issues

Source code: Content design that is not built in accordance with principles and norms might lead to software patches and technical issues.

Visitor access: Some platform creates room for chatting to promote visitor friendliness. They are highly vulnerable to attacks.

Security Software: It prevents the Web sites against cyber-attacks. Security as a service (SaaS) may be utilized to implement and manage security today.

Malware makes no distinctions: Adversary is not prejudiced. It does not differentiate between Web sites. Any Web site designed must ensure malware proof, and data present are secured as it builds reputation [9].

Increase of Security Attacks: There are many advanced methods practiced by hackers more often to attack Web sites. Malware may also be written to detect insecure Web pages and infect them.

4 Background and Analysis Work on Web Attacks and Its Statistics

The whole section describes numerous sorts of Web-based assaults, as well as how the attacker threatens the security of the system and some facts about Web vulnerabilities.

4.1 How Does an Attacker Impact Security?

Web sites are the primary cyberattack, who use Web applications as their primary base of operations and as their primary target for the majority of the Web sites they hack. Hackers use Web applications for a variety of reasons, including information warfare, blackmailing, serving malicious advertisements, and data theft including the theft of sensitive information like a person's credit card number or bank details and date of existence are just a few of the many reasons why hackers turn to online apps [10].

There are several steps involved in initiating a Web attack (Fig. 1).

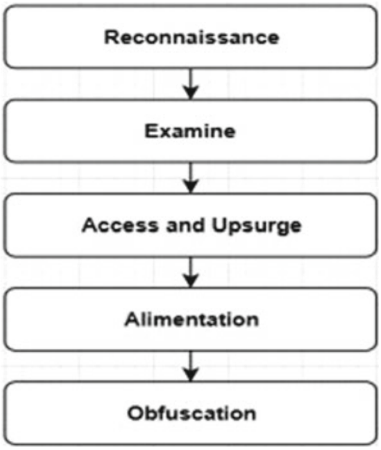


Fig. 1 Steps of web attacks

Reconnaissance: A successful online assault begins with a thorough examination and inspection of the target, followed by the determination of the most effective method of attack against that objective. Throughout each phase, they decide on a goal and a starting point for their observation. In this stage, the attacker just investigates their target in order to go further.

Examine: Following the discovery of the target, the attacker's primary goal is to identify the weakest link in order to proceed with their plan. They scan the whole system in an attempt to identify the point of entry into the company. This step of the assault is often lengthy, since it takes time to identify and exploit any vulnerabilities that have been discovered [10].

Access and Upsurge: It is necessary for the attacker to get proprietary information access to the system after finding an entry point. This is the most essential step in the lifecycle of a Web attack and should be done with considerable care and attention. Data may be stolen and used for malicious reasons if an intruder gains access to the system after getting highly classified access.

Alimentation: Once the attacker has gained access to the victim's most sensitive data, the next stage is to remain undetected in that system. Using root kits, hackers may now enter the system at any time and roam around freely, gaining access to all of the victim's private information and data. As a result of this, the intruder no longer needs to rely only on the weak entry point that they discovered previously, but may instead access the system at any time and from any location to carry out harmful operations utilizing the victim's credentials and sensitive data.

Obfuscation: To avoid detection, attackers try to hide their presence and the malware they are using by altering its general appearance. Assaults on computer systems are well-known, and anti-virus companies have developed sophisticated technologies for capturing the perpetrators. Many methods, like as spoofing and log cleaners, were used by the assailant to keep him from being discovered by the victim [10].

4.2 Web-Based Attacks Are Classified as Follows

We have seen a wide variety of assaults in this digital world, including network attacks, application attacks, and more. In this article, we will cover the most frequent Web-based assaults and how to avoid them. The following are the different types of attacks in Web Vulnerability: [11].

Authentication-Based Attacks: When a person enters into their account, the Web site runs an authentication procedure. Web sites check to see whether they are being accessed by an authorized user. An authentication-based attack may be successful if a Web site's verification process is ineffective. The authentication-based attacks are Brute force attack and credentials theft attacks.

Session Management: As the name suggests, it is a process that securely handle a large number of requests from a particular user or association of persons. A session is a sequence of HTTP requests and activities started by the particular user on the same Web site. To begin a session, users are often required to provide a password or some other kind of authentication method. The different attacks are: session hijacking, man in middle attack, and session replay [9].

Input validation attacks: For example, an attacker may purposefully provide a Web application an unexpected input in order to mislead it. To get access to the Web server, the attacker sends these odd inputs to the Web application and attempts to steal sensitive data from the database, such as usernames and password. The following are examples of input validation attacks: Buffer overflow attack, SQL injection, and cross site scripting [12].

Sensitive Data: In order to secure sensitive data from unauthorized access, it is necessary to implement appropriate safeguards. An individual user's relevant information, such as his or her identity, is classified as sensitive data. In order to avoid unwanted access and possible damage to the user, private data is occasionally used. The different attacks are data tempering and accessing sensitive data.

4.3 How to Handle These Risks?

1. Through Better Performance.
2. Consistent scanning and immediate malware removal: Security is ensured by regular and in-depth Web site scanning at the server level for malware.
3. Advanced security monitoring: The customer's or visitors private information is kept confidential and prevented from redirecting malicious Web sites that are highly infective to the system. For this purpose, Web site security relies on domain name system, secure socket layer, and WHOIS database [13].

4.4 Statistics

See Fig. 2.

4.5 The Frequency of Web-Based Attacks

According to recent Verizon research, Web applications are responsible for 43% of all breaches. However, applications are not the only source of vulnerability on the Web. Global search traffic has increased significantly over the last year, frequently spiking during periods of lockdown to combat the COVID-19 pandemic, as people

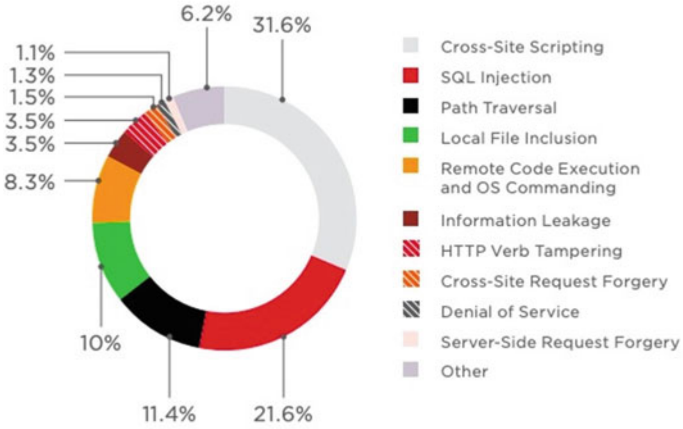


Fig. 2 Statistics of web attacks

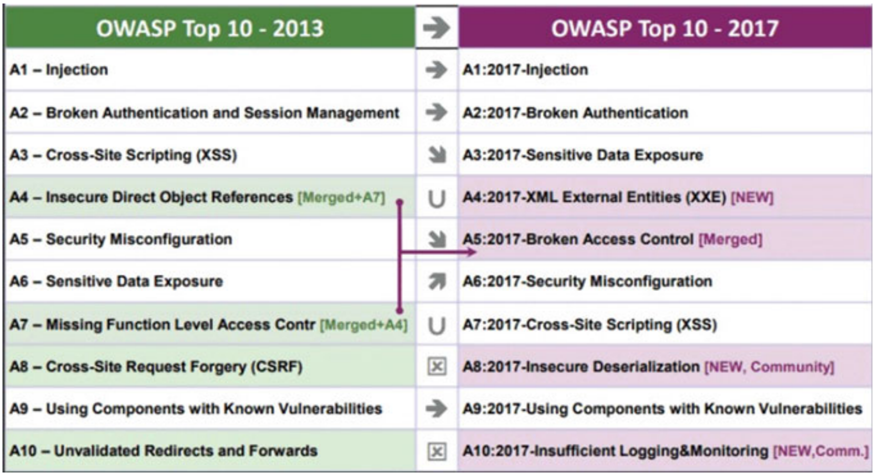
flocked to virtual hangouts and movie streaming platforms for entertainment and videoconferencing tools to communicate with coworkers remotely. Due to its reliance on the Internet, it was an attractive target for attackers, many of whom concentrated their efforts on Web vulnerabilities. According to SiteLock's analysis of 7 million Web sites, Web sites are currently subjected to an average of 94 attacks per day and are visited by bots approximately 2608 times per week [13].

5 OWASP [Open Web Application Security Project]

The OWASP is a non-profit and objected to Web application security. The OWASP top 10 is one of their most well-known projects, and it is available for free. Documentation, tools, videos, and discussion forums are some of the resources they provide (Figs. 3 and 4).

5.1 Top Web Security Risks

See Table 1.



OWASP Top 10 - 2013	→	OWASP Top 10 - 2017
A1 – Injection	→	A1:2017-Injection
A2 – Broken Authentication and Session Management	→	A2:2017-Broken Authentication
A3 – Cross-Site Scripting (XSS)	↘	A3:2017-Sensitive Data Exposure
A4 – Insecure Direct Object References [Merged+A7]	∪	A4:2017-XML External Entities (XXE) [NEW]
A5 – Security Misconfiguration	↘	A5:2017-Broken Access Control [Merged]
A6 – Sensitive Data Exposure	↗	A6:2017-Security Misconfiguration
A7 – Missing Function Level Access Contr [Merged+A4]	∪	A7:2017-Cross-Site Scripting (XSS)
A8 – Cross-Site Request Forgery (CSRF)	☒	A8:2017-Insecure Deserialization [NEW, Community]
A9 – Using Components with Known Vulnerabilities	→	A9:2017-Using Components with Known Vulnerabilities
A10 – Unvalidated Redirects and Forwards	☒	A10:2017-Insufficient Logging&Monitoring [NEW,Comm.]

Fig. 3 OWASP top 10 risk between 2013 and 2017

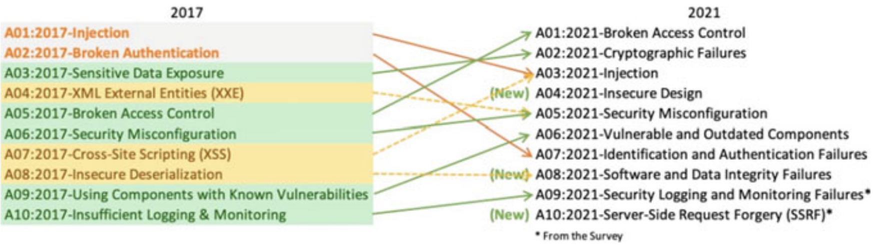

2017
A01:2017-Injection
A02:2017-Broken Authentication
A03:2017-Sensitive Data Exposure
A04:2017-XML External Entities (XXE)
A05:2017-Broken Access Control
A06:2017-Security Misconfiguration
A07:2017-Cross-Site Scripting (XSS)
A08:2017-Insecure Deserialization
A09:2017-Using Components with Known Vulnerabilities
A10:2017-Insufficient Logging & Monitoring
2021
A01:2021-Broken Access Control
A02:2021-Cryptographic Failures
A03:2021-Injection
(New) A04:2021-Insecure Design
A05:2021-Security Misconfiguration
A06:2021-Vulnerable and Outdated Components
A07:2021-Identification and Authentication Failures
(New) A08:2021-Software and Data Integrity Failures
A09:2021-Security Logging and Monitoring Failures*
(New) A10:2021-Server-Side Request Forgery (SSRF)*
* From the Survey

Fig. 4 OWASP top 10 risk between 2017 and 2021

Table 1 Web Security Vulnerabilities

• SQL injection • Password breach • Invalidated redirects and page forwards • Insecure direct object references • Code injection • Data breach • Security misconfiguration	• Remote file inclusion • Cross site request forgery (CSRF) • Broken authentication and session management • Insecure cryptographic storage • Failure to restrict URL access • Insufficient transport layer protection • Cross site scripting (XSS)

5.2 OWASP TOP 10 Vulnerabilities

Injection: When a Web application receives untrusted data as part of a command or query, an injection flaw such as SQL is exploited. The attacker's payload may trick the Web application into executing unwanted actions or gaining access to data without the user's knowledge or consent [1].

Broken Authentication: An intruder may get temporary or permanent access to other people's accounts by exploiting a weakness in the implementation of an application function that deals with the device authentication [14].

XML External Entities: Remote code execution and denial of service attacks may expose internal data to the public. External entities may be used to expose internal files to the broader public by using the file URI handler. Many outdated or poorly configured XML processors examine various transmission references included in XML documents.

Broken Access Control: Restrictions on what authorized users may do are often not imposed and enforced. These flaws may be exploited by hackers to get access to unauthorized functionality and/or information. Access restrictions can be changed, sensitive files can be accessed, and others can be altered.

Sensitive Data Exposure: If online applications fail to safeguard sensitive data such as financial information and passwords, attackers will be able to obtain access to that data and use it for their own malicious ends. An on-path attack is a common way of obtaining sensitive information from a computer system. The danger of data exposure may be reduced by encrypting any sensitive data and preventing the caching of any sensitive information, among other measures. Additionally, Web application developers should exercise caution to ensure that they are not keeping any sensitive data that is not absolutely necessary.

Security Misconfiguration: It is essential that all operating systems, guidelines, modules, and applications be set up securely, but they must always be modified on a regular basis to keep them safe and secure. Incorrect security settings are the most common cause of problems. Insecure default settings, insufficient or ad hoc configurations, and exposed cloud storage are all examples of this.

Cross Site Scripting: Allows an attacker to execute scripts in the browser of the victim. Users' sessions may be hijacked, Web sites can be defaced, or the victim might be sent to a hostile Web site. XSS attacks occur when an application does not properly validate or sanitize the data it sends to a new Web page.

Insecure Deserialization: Decoding problems may be used to initiate attacks like replay, insertion, and privilege escalation among others. Remote code execution is typically the consequence of insecure deserialization [14].

Components with Known Vulnerabilities: The libraries, structures, and other configuration files that make up an application have the same execution privileges as the applications themselves. Data loss or system takeover may occur if an exploit of a vulnerable component is successful. The defenses of applications and APIs may be compromised if they employ components that have known vulnerabilities.

Insufficient Logging and Monitoring: It is possible for attackers to continue to compromise a system, gain tenacity, and move on to other targets thanks to improper logging and analyzing as well as a lack of or inadequate interaction with incident response processes. It takes an average of over 200 days to discover a breach, according to most research, and external parties are more likely to do so than internal systems or monitoring Bhatt [15].

6 Conclusion

This article provides a thorough understanding of Web application security and the vulnerabilities that may be found in them, as well as a discussion of several OWASP weaknesses and their respective statistics. In order to employ safe coding methods to protect a system, every computer programmer and technologist must be aware of the most common dangers and the current status of cyber-attacks and data analysis. Finally, since Internet technology has advanced rapidly, the growth of Web-based application programs has been steady, and they have become an integral part of the contemporary computer platform. As Web technology has advanced at a breakneck pace, Web-based applications have grown clunky, resulting in the steady emergence of security flaws. Internet applications must be secured by completing a complete security analysis and designing an efficient, reliable technique of preventing assaults in order to assure the safety of programs and data.

References

1. Sobola TD, Zavarsky P, Butakov S (2020) Experimental study of modsecurity web application firewalls. Proceeding—2020 IEEE 6th International conference big data security cloud, BigDataSecurity 2020, 2020 IEEE International conference high perform smart computer HPSC 2020 2020 IEEE International conference intelligence data security IDS 2020, pp 209–213
2. Devi RS (2020)
3. Kumar S, Mahajan R, Kumar N, Khatri SK (2017) A study on web application security and detecting security vulnerabilities. 6th International conference reliability infocom technology optimization trends future directions ICRITO 2017, pp 451–455
4. Kong F (2017)
5. Geetha V, Kallapur PV (2011) Web security: research challenges and open issues. Lect Notes Electr Eng 121:397–404
6. Canfora G, Visaggio CA (2016) A set of features to detect web security threats. J Comput Virol Hacking Tech 12(4):243–261
7. Divyaniyadav D, Gupta D, Singh D, Kumar U, Sharma (2018) Vulnerabilities and security of web applications. Computer communication automation ICCCA, pp 1–5

8. Nunes P, Medeiros I, Fonseca JC, Neves N, Correia M, Vieira M (2018) Benchmarking static analysis tools for web security. IEEE Trans Reliab 67(3):1159–1175
9. Alenezi M, Nadeem M, Agrawal A, Kumar R, Khan RA (2020) Fuzzy multi criteria decision analysis method for assessing security design tactics for web applications. Int J Intell Eng Syst 13(5):181–196
10. Singh A, Sharma A, Sharma N, Kaushik I, Bhushan B (2019) Taxonomy of attacks on web based applications. International conference on intelligent computing, instrumentation and control technologies ICICICT 2019, pp 1231–1235
11. Touseef P (2019) Analysis of automated web application security vulnerabilities testing. PervasiveHealth Pervasive Comput Technol Healthc
12. Mishra S, Sharma SK, Alowaidi MA (2020) Analysis of security issues of cloud-based web applications. J Ambient Intell Humaniz Comput
13. Sadqi Y, Maleh Y (2021) A systematic review and taxonomy of web applica- tions threats. Inf Secur J 00(00):1–27
14. Higuera JRB, Higuera JB, Montalvo JAS, Villalba JC, Pérez JJN (2020) Benchmarking approach to compare web applications static analysis tools detecting OWASP top ten security vulnerabilities. Comput Mater Contin 64(3):1555–1577
15. Bhatt D (2018) Cyber security risks for modern web applications: case study paper for developers and security testers. Int J Sci Technol Res 7(5):232–235

Cloud Storage Client Forensic: Analysis of MEGA Cloud

Himanshu Mishra, Vikas Sihag, Gaurav Choudhary, Nicola Dragoni, and Ilsun You

Abstract In the current decade, the use of cloud storage services is drastically increasing because it provides large data storage with various functionalities. Consequently, it is easier for someone to download, upload, change, and share data on multiple platforms. Meanwhile, cloud storage forensics is of paramount importance for service providers to rectify security issues. However, the unique characteristics of cloud service introduce numerous challenges like resource pooling, on-demand services, investigation, and legal matters. Cloud storage forensic deals with phenomena involving the cloud infrastructure and corresponding services in civil proceedings and criminal investigations. This paper presents a forensic analysis of the MEGA, a cloud service provider, using the Windows 10 operating system and android platform. In this investigation, we recognize various artifacts such as uploading, log in, downloading, file timestamps, and file sharing that could be forensically recovered.

Keywords Cloud storage forensics · Windows browser based · Windows application based · Android application based

H. Mishra · V. Sihag (✉)
Security and Criminal Justice, Sardar Patel University of Police, Jodhpur, India
e-mail: vikas.sihag@policeuniversity.ac.in

H. Mishra
e-mail: spu19cshm@policeuniversity.ac.in

G. Choudhary · N. Dragoni
DTU Compute, Department of Applied Mathematics and Computer Science, Technical University of Denmark (DTU), Lyngby, Denmark
e-mail: gauravchoudhary7777@gmail.com

N. Dragoni
e-mail: ndra@dtu.dk

I. You
Department of Information Security Engineering, Soonchunhyang University, Asan, Republic of Korea
e-mail: ilsunu@gmail.com

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_79

1 Introduction

In cloud storage services, subscribers can store data, including but not limited to music, videos, and pictures files. It allows users to access their files using a client cloud storage application or via a Web portal on any platform such as smartphones, personal computers, and tablets. [10]. Furthermore, many organizations today are employing cloud services for all their digital needs (e.g., platform, system, and software). Unfortunately, the increasing number of migration to cloud storage services also becomes the hotbed for criminal activities. In this case, forensic analysis of cloud storage services becomes more complex than traditional computer forensics because of its unique characteristics. It requires investigators to identify, collect, and examine the data from multiple devices used to access the files (e.g., PC, smartphone, etc.) [1]. The bits of collected evidence during the investigation are crucial since they are considered user artifacts. Additionally, the advancement of virtualization technologies makes the collection of user activity logs more challenging. However, in cloud-based businesses, the data are controlled by a third party, storing all the data in one place. Such a drawback can be exploited for forensic investigations.

This investigative research analyzes MEGA cloud storage service, a popular cloud storage as a service (STaaS) provider. The users of MEGA cloud may upload, download, and access their data using the MEGA client application and Web browser (e.g., Google Chrome) using any computer and smartphone. Clients of MEGA cloud can do file hosting, sync, store, and share their data. The MEGA provides to their clients a storage space for free of up to 50GB. Table 1 presents the features of MEGA.

In this paper, we investigate the kind of artifacts residing in disk and memory after usage of MEGA cloud services from (a) Google Chrome(GC) browser in Windows, (b) Windows client application in Windows 10, and (c) Android app.

The remainder of this paper is structured as follows. Section 2 discusses the related work on different cloud storage service providers. The research methodology setup is covered in Sect. 3, while Sect. 4 introduces the adopted framework for the investigation. Section 5 presents evidence analysis and findings based on the experiment on different applications. Finally, Sect. 6 concludes this paper.

Table 1 Features of MEGA

MEGA cloud		
Operating system support	Windows	Yes
	Linux	Yes
	Mac OS	Yes
	iOS	Yes
	Android	Yes
Storage free		50 GB
Backup		Yes
Sync		Yes
Encryption		Yes
Sharing		Yes

2 Related Works

Ko et al. [5] conducted an experimental investigation with Google Drive, in which investigators identified the locations of related artifacts. The recovered residual artifacts, which underwent further examination, include thumbnails, images, metadata, contacts, system logs, and some temporary files. The metadata files were stored in the tabular form and their attributes (such as creation time, modification time, sync time, file name, and identifiers) in their analysis. The authors stored the metadata files in the tabular form based on their attributes (e.g., creation time, modification time, sync time, file name, and identifiers). In addition, they recovered the contents of deleted files by examining thumbnails from the cache directory. Hale [9] discusses the artifacts left behind after using Amazon Cloud Drive on the client computer. Through analyzing the browser history files and Web browser cache files, the investigator identified and collected valuable fields such as Amazon customer ID, file name, cloud path, file creation date, last update date, file name, and object ID. To maintain the integrity of the data, the investigator checks the MD5 hash value with the original data. In analyzing the registry artifacts, the investigator finds the location of the installation of the Amazon Cloud Drive application.

Ko et al. in [6] performs analysis of the OneDrive application on the iOS platform. The investigators collected the artifacts from the TarArchive folder. The authors highlighted that investigators should examine cache and thumbnail remnants such as images, word documents, audio, videos, and PDF files for forensic evidence. In the experiment, the investigators discovered that the OneDrive could not directly share the document but first needed to download the shareable link in recovering documents. Daryabar et al. [3] proposed an investigation on the MEGA cloud client application installed on the iOS and android platforms. To implement the experiment, the investigator used the EDRM Enron file format dataset. The investigator examined the impact of evidence preservation when metadata and file content modification occurs during the download and upload process on iOS and android platforms. Additionally, the investigators also recovered potential forensic and network traffic artifacts on both platforms. Furthermore, the investigator also identified users activities artifacts such as deletion of files, login, timestamp in UNIX format, and sharing files.

Martini et al. [8] proposed an investigation on open-source cloud StaaS application, known as ownCloud, installed on the CentOS operating system. The investigator successfully conducted a forensic examination of the server and client components. In the investigation of client forensic evidence using the Windows 7 environment, the investigator collected a range of forensic artifacts such as cached files, synced files, sync metadata, timestamp which in UNIX format, file management metadata, cloud service data, authentication data, their browser artifacts, URL parameters, and mobile client artifacts or network data. Shariatic et al. [11] proposed an investigation on the cloud StaaS application SugarSync client application installed in the Mac OS operating system, Windows 8 operating system, android, and iOS-based devices. In the examination of the Windows browser and application, the investigators collected from the Internet Explorer (IE), Google Chrome (GC), Firefox (FX), Apple

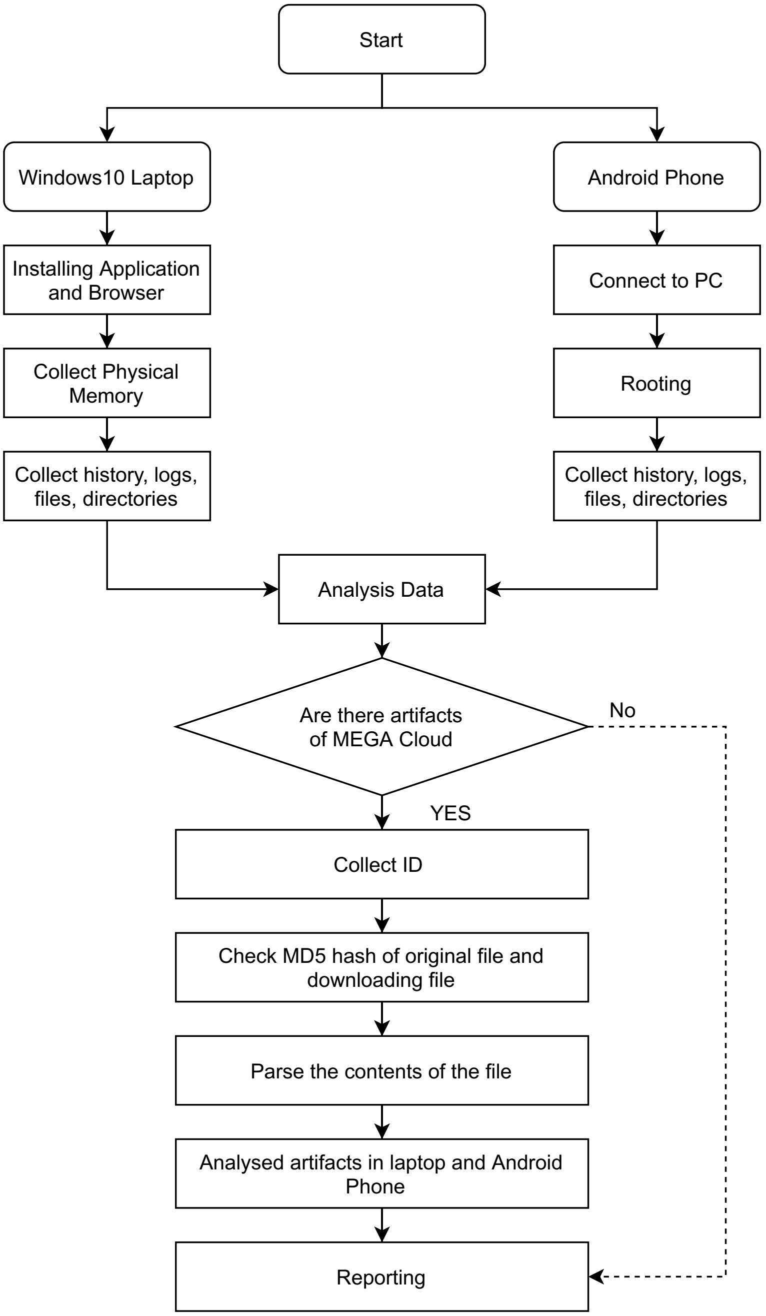

Start
Windows10 Laptop
Android Phone
Installing Application and Browser
Connect to PC
Collect Physical Memory
Rooting
Collect history, logs, files, directories
Collect history, logs, files, directories
Analysis Data
Are there artifacts of MEGA Cloud
No
YES
Collect ID
Check MD5 hash of original file and downloading file
Parse the contents of the file
Analysed artifacts in laptop and Android Phone
Reporting

Fig. 1 The workflow of an investigation procedure in a MEGA

Table 2 Different files used for analysis during experiments

Name	MD5	Note
File1.jpg	0ADC4258C90CFF58C2909CE560D637FE	A jpg extension file
File2.png	0721A7F4734F2C0AC39B2EFBBD45E11F	A png extension file
File3.docx	9D80E507516F4DAFC19B93775C93D0D2	A docx extension file
File4.pdf	CAF527FBE7F500B9F9402C306E753642	A pdf extension file
File5.mp3	7DE4686F0A96AF09A8B404B665F11B6C	A mp3 extension file
File6.mp4	3B6B25CC6623083D005DE592792AA678	A mp4 extension file
File7.jpg	12BFED37A17A0C74FF2DA0D234B47C30	A jpg extension file
File8.png	D0ACF757945B07B6CFCFF7B8E0D9B0FF	A png extension file
File9.docx	09AF4B043BD3E0C8B9CF097CC79AE98F	A docx extension file
File10.pdf	3C00528B826796DF30D9C97FCE3E8733	A pdf extension file
File11.mp3	3D341FF78071FC354D7C12327ACC7813	A mp3 extension file
File12.mp4	4F7D6F0088180C913881DF5F17306F50	A mp4 extension file

Safari (AS) a range of artifacts such as cache, history of browser, timestamp based on UNIX format, upload, live memory, dataset, registry, file system, log files, username, password, user id, and network traffic.

3 System Model and Methodology

The goal of the paper is to investigate Mega cloud service storage for forensic evidence. Accordingly, we followed a set of precise procedures shown in Fig. 1. In this experiment, we used 12 different files comprising of images, document, audio, and video as shown in Table 2 with their respective MD5 hash. Files 1–6 are used for Windows, whereas 7–12 for android. In performing the investigation, the keywords, including *MEGA*, *login*, *uploading*, *downloading*, and *email*, are used to find valuable remnants.

4 Experimental Setup and Case Scenario

In this investigation, we concert the four steps of the cloud forensics model in testing MEGA cloud features on Windows and android: identification, collection, preservation, examination, and analysis. Additionally, all MEGA-related applications were downloaded from the Mega Web site and corresponding mobile stores. The scenario is described as follows using the cloud forensic model.

- *Identification and collection*: In the android platform, the smartphone Redmi 6 device was used for evidence collection, whereas a Windows 10 operating was used to investigate desktop applications. The network traffic of the browser and application was also monitored using the Wireshark tool.

- *Preservation*: The MD5 hash values of uploaded and downloaded files were calculated to check file integrity.
- *Examination and Analysis*: Possible data remnants of MEGA application like internal memory, storage, and network traffic capture backups were identified for further examination and analysis of user artifacts (such as sharing, upload, download, and login information).

5 Results and Analysis

In this section, findings during the investigation of the collected data from android phones and Windows 10-based Laptops are listed. Accordingly, we have recovered informative evidence linked with the MEGA client user using the Google Chrome browser version 89.0.4389.114 and the MEGA client application. Tools used for investigative purposes are listed in Table 3. The investigation procedure was repeated twice at a different date to validate and ensure the consistency of evidence findings.

5.1 Google Chrome Browser-Based Artifacts

While accessing MEGA services using Google Chrome, we recovered various files and browser artifacts, along with paths, as shown in Table 4. Artifacts are examined and evaluated in a step-by-step process. The evaluated data sources are as follows:

Dataset: The data related to the upload and download operations were recovered from the memory. The summary of uploaded and downloaded filenames with

Table 3 Tools used for MEGA cloud forensic

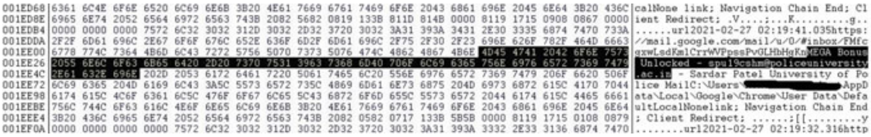
Tools	Usage
File Alyzer 2.05.57	To extract file details like hash, date, time etc.
Hex workshop v5	To search keyword in the forensic images
WinMD5	To check the MD5 hash value of an original and downloaded file
SQLite Forensic Explorer v 2.0	To view SQLite database file content
Hindsight-master	To analyze Google chrome artifacts
Winprefetchview v 1.36	To analyze the Windows prefetch files
Wireshark 3.4.4	To analyze network traffic
Adb tools	To analyze and connect the android phone with pc
Mobiledit v 7.2.0.17975	To analyze application data of the android phone
Fridump v0.1	To take android RAM dump

Table 4 Artifact locations while performing browser forensics.

Artifact	Path
Cache	`C:Users\UserName\AppData\Local\Google\Chrome\User Data\Default\Cache`
History	`C:Users\UserName\AppData\Local\Google\Chrome\User Data\Default\History`
Cookies	`C:Users\UserName\AppData\Local\Google\Chrome\User Data\Default\Cookies`
Login details	`C:Users\UserName\AppData\Local\Google\Chrome\User Data\Default\Login Data`
IndexedDB	`C:Users\UserName\AppData\Local\Google\Chrome\User Data\Default\IndexedDB\https\_mega.nz\_0.indexeddb.leveldb`
Session	`C:Users\UserName\AppData\Local\Google\Chrome\User Data\Default\Session Storage`
Download	`C:Users\UserName\Downloads`

Table 5 Artifacts related to uploaded and downloaded files using browser

Filename	Type	Date/Time	MD5 hash
File1.jpg	Original	2021-02-07 15:11:54	0ADC4258C90CFF58C2909CE560D637FE
	Downloaded	2021-02-08 13:02:05	0ADC4258C90CFF58C2909CE560D637FE
File2.png	Original	2021-02-13 17:12:29	0721A7F4734F2C0AC39B2EFBBD45E11F
	Downloaded	2021-02-14 09:02:15	0721A7F4734F2C0AC39B2EFBBD45E11F



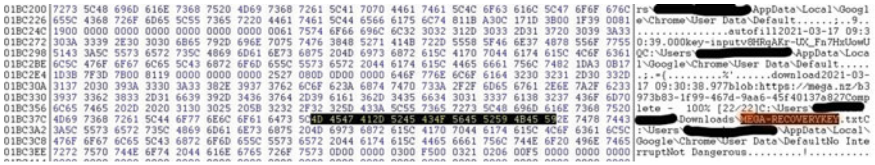
```
001ED68|6361 6C4E 6F6E 6520 6C69 6E6B 3B20 4E61 7669 6761 7469 6F6E 2043 6861 696E 2045 6E64 3B20 436C|calNone link; Navigation Chain End; Cl
001ED8E|6965 6E74 2052 6564 6972 6563 743B 2082 5682 0819 133B 811D 814B 0000 8119 1715 0908 0867 0000|ient Redirect; .V....;...K.........g..
001EDB4|0000 0000 0000 7572 6C32 3032 312D 3032 2D32 3720 3032 3A31 393A 3431 2E30 3335 6874 7470 733A|......url2021-02-27 02:19:41.035https:
001EDDA|2F2F 6D61 696C 2E67 6F6F 676C 652E 636F 6D2F 6D61 696C 2F75 2F30 2F23 696E 626F 782F 464D 6663|//mail.google.com/mail/u/0/#inbox/FMfc
001EE00|6778 774C 7364 4B6D 6C43 7272 5756 5070 7373 5076 474C 4862 4867 4B6E 4D45 4741 2042 6F6E 7573|gxwLsdKmlCrrWVPpssPvGLHbHgKnMEGA Bonus
001EE26|2055 6E6C 6F63 6B65 6420 2D20 7370 7531 3963 7368 6D40 706F 6C69 6365 756E 6976 6572 7369 7479| Unlocked - spu19cshm@policeuniversity
001EE4C|2E61 632E 696E 202D 2053 6172 6461 7220 5061 7465 6C20 556E 6976 6572 7369 7479 206F 6620 506F|.ac.in - Sardar Patel University of Po
001EE72|6C69 6365 204D 6169 6C43 3A5C 5573 6572 735C 4869 6D61 6E73 6875 204D 6973 6872 615C 4170 7044|lice MailC:\Users\[redacted]AppD
001EE98|6174 615C 4C6F 6361 6C5C 476F 6F67 6C65 5C43 6872 6F6D 655C 5573 6572 2044 6174 615C 4465 6661|ata\Local\Google\Chrome\User Data\Defa
001EEBE|756C 744C 6F63 616C 4E6F 6E65 6C69 6E6B 3B20 4E61 7669 6761 7469 6F6E 2043 6861 696E 2045 6E64|ultLocalNonelink; Navigation Chain End
001EEE4|3B20 436C 6965 6E74 2052 6564 6972 6563 743B 2082 0582 0717 133B 5B5B 0000 8119 1715 0108 0879|; Client Redirect; ......;[[.........y
001EF0A|0000 0000 0000 0000 7572 6C32 3032 312D 3032 2D32 3720 3032 3A31 393A 3332 2E33 3136 6874 7470|........url2021-02-27 02:19:32.316http
```

Fig. 2 User identifier located during memory analysis

respective hashes and timestamps is listed in Table 5. In some cases, the timestamp is defined in UNIX time format.

Network analysis: The communication session between the Google Chrome browser and the MEGA cloud server was established through TLSv1.2 with an IP address ipv4.scr 66.163.35.36 and ipv6.scr 2a0b:e46:1:100::11 on port no 443.

Memory analysis: When searching credentials through memory analysis, as shown in Fig. 2, we were able to locate a unique identifier of MEGA bonus unlocked rather than the MEGA credentials utilized in the investigation. Our analysis discovered that a substitute field descriptor, such as login-name2, uses a similar identification value.



```
01BC200|7273 5C48 696D 616E 7368 7520 4D69 7368 7261 5C41 7070 4461 7461 5C4C 6F63 616C 5C47 6F6F 676C|rs\[redacted]AppData\Local\Googl
01BC226|655C 4368 726F 6D65 5C55 7365 7220 4461 7461 5C44 6566 6175 6C74 811B A30C 171D 3B00 1F39 0081|e\Chrome\User Data\Default......;..9..
01BC24C|1900 0000 0000 0000 0000 0000 0000 0061 7574 6F66 696C 6C32 3032 312D 3033 2D31 3720 3039 3A33|..................autofill2021-03-17 09:3
01BC272|303A 3339 2E30 3030 6B65 792D 696E 7075 7476 3848 5271 414B 722D 5558 5F46 6E37 4878 556F 7755|0:39.000key-inputv8HRqAKr-UX_Fn7HxUowU
01BC298|5143 3A5C 5573 6572 735C 4869 6D61 6E73 6875 204D 6973 6872 615C 4170 7044 6174 615C 4C6F 6361|QC:\Users\[redacted]AppData\Loca
01BC2BE|6C5C 476F 6F67 6C65 5C43 6872 6F6D 655C 5573 6572 2044 6174 615C 4465 6661 756C 7482 1DA3 0B17|l\Google\Chrome\User Data\Default.....
01BC2E4|1D3B 7F3D 7B00 8119 0000 0000 0000 2527 080D 0D00 0000 646F 776E 6C6F 6164 3230 3231 2D30 332D|.;.=(..........%'......download2021-03-
01BC30A|3137 2030 393A 3330 3A33 382E 3937 3762 6C6F 623A 6874 7470 733A 2F2F 6D65 6761 2E6E 7A2F 6233|17 09:30:38.977blob:https://mega.nz/b3
01BC330|3937 3362 3833 2D31 6639 392D 3436 3764 2D39 6161 362D 3435 6634 3031 3337 6138 3237 436F 6D70|973b83-1f99-467d-9aa6-45f40137a827Comp
01BC356|6C65 7465 202D 2020 3130 3025 205B 3232 2F32 325D 433A 5C55 7365 7273 5C48 696D 616E 7368 7520|lete -  100% [22/22]C:\Users\[redacted]
01BC37C|4D69 7368 7261 5C44 6F77 6E6C 6F61 6473 5C4D 4547 412D 5245 434F 5645 5259 4B45 592E 7478 7443|[redacted]Downloads\MEGA-RECOVERYKEY.txtC
01BC3A2|3A5C 5573 6572 735C 4869 6D61 6E73 6875 204D 6973 6872 615C 4170 7044 6174 615C 4C6F 6361 6C5C|:\Users\[redacted]\AppData\Local\
01BC3C8|476F 6F67 6C65 5C43 6872 6F6D 655C 5573 6572 2044 6174 615C 4465 6661 756C 744E 6F20 496E 7465|Google\Chrome\User Data\DefaultNo Inte
01BC3EE|7272 7570 744E 6F74 2044 616E 6765 726F 7573 0D00 0000 0300 F500 0321 0206 00F5 0000 0000 0000|rruptNot Dangerous..........!..........
```

Fig. 3 User file artifacts identified during memory analysis

Table 6 Artifacts related to uploaded and downloaded files using MEGA Windows client application

Filename	Type	Date/Time	MD5 hash
File5.mp3	Original	2021-02-07 15:14:32	7DE4686F0A96AF09A8B404B665F11B6C
	Downloaded	2021-02-08 13:02:42	7DE4686F0A96AF09A8B404B665F11B6C
File6.mp4	Original	2021-02-07 15:15:05	3B6B25CC6623083D005DE592792AA678
	Downloaded	2021-02-08 13:02:58	3B6B25CC6623083D005DE592792AA678

Upload and Download: Multiple times during the investigation, we located the name of the downloaded file in the memory, as shown in Fig. 3. File information related to user activity is vital information from the perspective of the investigator.

Cache and Browser history: The cache and browser history were also analyzed using the Hindsight-master tool. Accordingly, we were able to recover a indication of successful login to the MEGA through the URL: https://g.api.mega.co.nz/wsc?id=452920252&sid=7zNVuzrzDPUUpxEaFRkah1p5SGNZeHIwdW5N1cZZtKpeFRPYd93-rGboNw&ec&domain=meganz&ut=_MJ_nd2XvDw&v=2&lang=en&sn=YyaKGxsXdEI. However, we could not recuperate the data like username and secret key. Despite this, we can still observe that one could access the MEGA record without any certification requirement if the 'Keep me signed in' was chosen.

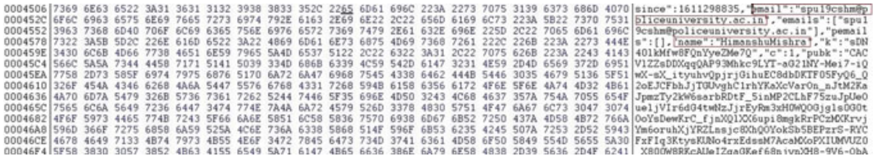
5.2 *Windows Application-Based Forensics*

As designated earlier, MEGA application permits a user to upload files up to 50GB for free, and synchronize any folder on the user devices. During installation of the application, a folder named MEGAsync Downloads is automatically created at the default path `C:\Users\UserName\Documents\MEGAsync\Downloads`. The folder can be opened and accessed even if the application is offline. In addition, a MEGA Drive is also created at the location `C:\Users\UserName\AppData\Local\Mega Limited`. Meanwhile, location `C:\Users\$users$\AppData\Local\MEGAsync`stores logs, database, .dll and other files associated to MEGA [2]. **Dataset** We recovered data related to the upload and download operations in

the memory using the MEGA Windows client application. A summary of the file artifacts is given in Table 6. In some cases, the timestamp is in UNIX format.

Table 7 Mega windows client application artifacts

Location	Artifacts collected
Megasync.version	Mega application installed version
MEGAsync.cfg	Mega application configuration
MEGAsync.log	Network information
	Timeout information
	User-Agent
	Cryptopp version
	MediaInfo version
	public key hash
	DB transaction
	Email
	name



```
0004506|7369 6E63 6522 3A31 3631 3132 3938 3833 352C 2265 6D61 696C 223A 2273 7075 3139 6373 686D 4070|since":1611298835,"email":"spu19cshm@p
000452C|6F6C 6963 6575 6E69 7665 7273 6974 792E 6163 2E69 6E22 2C22 656D 6169 6C73 223A 5B22 7370 7531|oliceuniversity.ac.in","emails":["spu1
0004552|3963 7368 6D40 706F 6C69 6365 756E 6976 6572 7369 7479 2E61 632E 696E 225D 2C22 7065 6D61 696C|9cshm@policeuniversity.ac.in"],"pemail
0004578|7322 3A5B 5D2C 226E 616D 6522 3A22 4869 6D61 6E73 6875 4D69 7368 7261 222C 226B 223A 2273 444E|s":[],"name":"HimanshuMishra","k":"sDN
000459E|3430 6C6B 4D66 7738 4651 6E59 7965 5A4D 6537 5122 2C22 6322 3A31 2C22 7075 626B 223A 2243 4143|40lkMfw8FQnYyeZMe7Q","c":1,"pubk":"CAC
00045C4|566C 5A5A 7344 4458 7171 5141 5039 334D 686B 6339 4C59 542D 6147 3231 4E59 2D4D 6569 372D 6951|VlZZsDDXqqQAP93Mhkc9LYT-aG21NY-Mei7-iQ
00045EA|7758 2D73 585F 6974 7975 6876 5170 6A72 6A47 6968 7545 4338 6462 444B 5446 3035 4679 5136 5F51|wX-sX_ityuhvQpjrjGihuEC8dbDKTF05FyQ6_Q
0004610|326F 454A 4346 6268 4A6A 5447 5576 6768 4331 7268 594B 6158 6356 6172 4F6E 5F6E 4A74 4D32 4B61|2oEJCFbhJjTGUvghC1rhYKaXcVarOn_nJtM2Ka
0004636|4A70 6D7A 5479 326B 5736 7361 7262 5244 7446 5F35 696E 4D50 3243 4C68 4637 357A 754A 7055 654F|JpmzTy2kW6sarbRDtF_5inMP2CLhF75zuJpUeO
000465C|7565 6C6A 5649 7236 6447 3474 774E 7A4A 6A72 4579 526D 3378 4830 5751 4F47 6A67 6C73 3047 3074|ueljVIr6dG4twNzJjrEyRm3xH0WQOGjgls0G0t
0004682|4F6F 5973 4465 774B 7243 5F66 6A6E 5851 6C58 5836 7570 6938 6D67 6B52 7250 437A 4D58 4B72 766A|OoYsDewKrC_fjnXQlXX6upi8mgkRrPCzMXKrvj
00046A8|596D 366F 7275 6858 6A59 525A 4C6E 736A 6338 5868 514F 596F 6B53 6235 4245 507A 7253 2D52 5943|Ym6oruhXjYRZLnsjc8XhQOYokSb5BEPzrS-RYC
00046CE|4678 4649 7133 4B74 7973 4B55 4E6F 3472 7845 6473 734D 3741 6361 4D58 6F50 5849 554D 5655 5A30|FxFIq3KtysKUNo4rxEdssM7AcaMXoPXIUMVUZ0
00046F4|5F58 3830 3057 3852 4B63 4155 6549 5A71 6147 4B65 6636 386E 6A79 6E58 4838 2D39 5636 2D4F 6241|_X800W8RKcAUeIZqaGKef68njynXH8-9V6-ObA
```

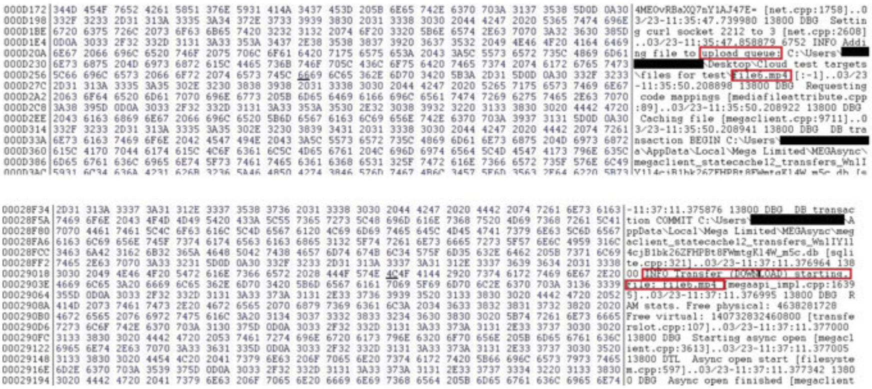
Fig. 4 Email and name located in memory

Network analysis The communication session between the client application and the cloud server was established through TLSv1.2 with an IP address ipv4.scr 66.203.125.11 and ipv6.scr 2a01:111:202c::200 on port no 443, respectively.

Memory Memory analysis enables us to locate artifacts related to the user. In addition, we were also able to discover timestamps in UNIX format, which helped identify the utilization time of the applications. Moreover, we also collected artifacts such as network information, timeout information, used cryptopp version, their user agent, public key hash, and DB transaction that are stored in the log file. Besides, we also identified the Megasync.version and MEGAsync.cfg files as shown in Table 7 (Figs. 4 and 5).

Upload and Download Through analysis of the MEGAsync.log file, the uploaded file location and time information were identified. We also recovered the download information such as file name and time at `C:\Users\Username\Documents Downloads`.

Prefetch files The prefetch file has a unicode list of executable DLLs files, the timestamp showing the last utilization time of the MEGA application, the execution count of the executable file, and the detailed information of the executable program. In our investigation, the identified prefetch files were MEGASYNC.EXE-98D4622B.pf and MEGASYNCSETUP64.EXE-1C8F9B1F.pf.



```
000D172|344D 454F 7652 4261 5851 376E 5931 414A 3437 453D 205B 6E65 742E 6370 703A 3137 3538 5D0D 0A30|4MEOvRBaXQ7nY1AJ47E= [net.cpp:1758]..0
000D198|332F 3233 2D31 313A 3335 3A34 372E 3733 3939 3830 2031 3338 3030 2044 4247 2020 5365 7474 696E|3/23-11:35:47.739980 13800 DBG  Settin
000D1BE|6720 6375 726C 2073 6F63 6B65 7420 3232 3132 2074 6F20 3320 5B6E 6574 2E63 7070 3A32 3630 385D|g curl socket 2212 to 3 [net.cpp:2608]
000D1E4|0D0A 3033 2F32 332D 3131 3A33 353A 3437 2E38 3538 3837 3920 3637 3532 2049 4E46 4F20 4164 6469|..03/23-11:35:47.858879 6752 INFO Addi
000D20A|6E67 2066 696C 6520 746F 2075 706C 6F61 6420 7175 6575 653A 2043 3A5C 5573 6572 735C 4869 6D61|ng file to upload queue: C:\Users\[redacted]
000D230|6E73 6875 204D 6973 6872 615C 4465 736B 746F 705C 436C 6F75 6420 7465 7374 2074 6172 6765 7473|\Desktop\Cloud test targets
000D256|5C66 696C 6573 2066 6F72 2074 6573 745C 6669 6C65 362E 6D70 3420 5B3A 2D31 5D0D 0A30 332F 3233|\files for test\file6.mp4 [:-1]..03/23
000D27C|2D31 313A 3335 3A35 302E 3230 3838 3938 2031 3338 3030 2044 4247 2020 5265 7175 6573 7469 6E67|-11:35:50.208898 13800 DBG  Requesting
000D2A2|2063 6F64 6520 6D61 7070 696E 6773 205B 6D65 6469 6166 696C 6561 7474 7269 6275 7465 2E63 7070| code mappings [mediafileattribute.cpp
000D2C8|3A38 395D 0D0A 3033 2F32 332D 3131 3A33 353A 3530 2E32 3038 3932 3220 3133 3830 3020 4442 4720|:89]..03/23-11:35:50.208922 13800 DBG 
000D2EE|2043 6163 6869 6E67 2066 696C 6520 5B6D 6567 6163 6C69 656E 742E 6370 703A 3937 3131 5D0D 0A30| Caching file [megaclient.cpp:9711]..0
000D314|332F 3233 2D31 313A 3335 3A35 302E 3230 3839 3431 2031 3338 3030 2044 4247 2020 4442 2074 7261|3/23-11:35:50.208941 13800 DBG  DB tra
000D33A|6E73 6163 7469 6F6E 2042 4547 494E 2043 3A5C 5573 6572 735C 4869 6D61 6E73 6875 204D 6973 6872|nsaction BEGIN C:\Users\[redacted]
000D360|615C 4170 7044 6174 615C 4C6F 6361 6C5C 4D65 6761 204C 696D 6974 6564 5C4D 4547 4173 796E 635C|a\AppData\Local\Mega Limited\MEGAsync\
000D386|6D65 6761 636C 6965 6E74 5F73 7461 7465 6361 6368 6531 325F 7472 616E 7366 6572 735F 576E 6C49|megaclient_statecache12_transfers_WnlI
000D3AC|5931 6C34 636A 4231 626B 3236 5A46 4850 4274 3846 576D 7467 4B6C 3457 5F6D 3563 2E64 6220 5B73|Y1l4cjB1bk26ZFHPBt8FWmtgKl4W_m5c.db [s
```

```
00028F34|2D31 313A 3337 3A31 312E 3337 3538 3736 2031 3338 3030 2044 4247 2020 4442 2074 7261 6E73 6163|-11:37:11.375876 13800 DBG  DB transac
00028F5A|7469 6F6E 2043 4F4D 4D49 5420 433A 5C55 7365 7273 5C48 696D 616E 7368 7520 4D69 7368 7261 5C41|tion COMMIT C:\Users\[redacted]\A
00028F80|7070 4461 7461 5C4C 6F63 616C 5C4D 6567 6120 4C69 6D69 7465 645C 4D45 4741 7379 6E63 5C6D 6567|ppData\Local\Mega Limited\MEGAsync\meg
00028FA6|6163 6C69 656E 745F 7374 6174 6563 6163 6865 3132 5F74 7261 6E73 6665 7273 5F57 6E6C 4959 316C|aclient_statecache12_transfers_WnlIY1l
00028FCC|3463 6A42 3162 6B32 365A 4648 5042 7438 4657 6D74 674B 6C34 575F 6D35 632E 6462 205B 7371 6C69|4cjB1bk26ZFHPBt8FWmtgKl4W_m5c.db [sqli
00028FF2|7465 2E63 7070 3A33 3231 5D0D 0A30 332F 3233 2D31 313A 3337 3A31 312E 3337 3639 3634 2031 3338|te.cpp:321]..03/23-11:37:11.376964 138
00029018|3030 2049 4E46 4F20 5472 616E 7366 6572 2028 444F 574E 4C4F 4144 2920 7374 6172 7469 6E67 2E20|00 INFO Transfer (DOWNLOAD) starting. 
0002903E|4669 6C65 3A20 6669 6C65 362E 6D70 3420 5B6D 6567 6161 7069 5F69 6D70 6C2E 6370 703A 3136 3339|File: file6.mp4 [megaapi_impl.cpp:1639
00029064|355D 0D0A 3033 2F32 332D 3131 3A33 373A 3131 2E33 3736 3939 3520 3133 3830 3020 4442 4720 2052|5]..03/23-11:37:11.376995 13800 DBG  R
0002908A|414D 2073 7461 7473 2E20 4672 6565 2070 6879 7369 6361 6C3A 2034 3633 3832 3831 3732 3820 2020|AM stats. Free physical: 4638281728  
000290B0|4672 6565 2076 6972 7475 616C 3A20 3134 3037 3332 3833 3234 3630 3830 3020 5B74 7261 6E73 6665|Free virtual: 140732832460800 [transfe
000290D6|7273 6C6F 742E 6370 703A 3130 375D 0D0A 3033 2F32 332D 3131 3A33 373A 3131 2E33 3737 3030 3020|rslot.cpp:107]..03/23-11:37:11.377000 
000290FC|3133 3830 3020 4442 4720 2053 7461 7274 696E 6720 6173 796E 6320 6F70 656E 205B 6D65 6761 636C|13800 DBG  Starting async open [megacl
00029122|6965 6E74 2E63 7070 3A33 3631 335D 0D0A 3033 2F32 332D 3131 3A33 373A 3131 2E33 3737 3030 3520|ient.cpp:3613]..03/23-11:37:11.377005 
00029148|3133 3830 3020 4454 4C20 2041 7379 6E63 206F 7065 6E20 7374 6172 7420 5B66 696C 6573 7973 7465|13800 DTL  Async open start [filesyste
0002916E|6D2E 6370 703A 3539 375D 0D0A 3033 2F32 332D 3131 3A33 373A 3131 2E33 3737 3334 3220 3133 3830|m.cpp:597]..03/23-11:37:11.377342 1380
00029194|3020 4442 4720 2041 7379 6E63 206F 7065 6E20 6669 6E69 7368 6564 205B 6D65 6761 636C 6965 6E74|0 DBG  Async open finished [megaclient
```

Fig. 5 Memory analysis: uploaded and downloaded files

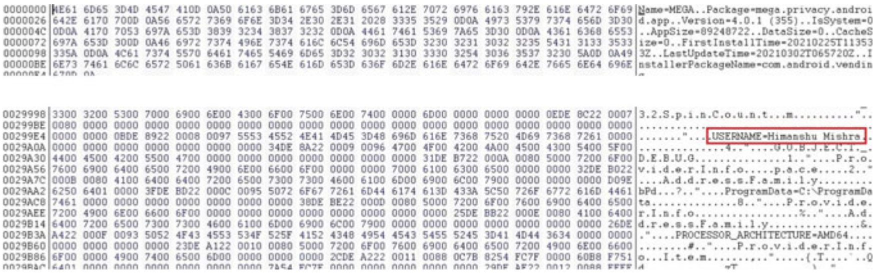
Registry MEGA application installation of Windows adds registry hives in following locations:

- `HKEY_CLASSES_ROOT\Drive\shellex\ContextMenuHandlers\MEGA (Context menu)`
- `HKEY_CLASSES_ROOT\LocalSettings\Software\Microsoft\Windows\CurrentVersion\AppModel\SystemAppData\Microsoft.Windows.Photos_8wekyb3d8bbwe\PersistedStorageItemTable\ManagedByApp\9C9BD299-C3FB-4A5F-8F6B-CA97A45BE0B6`
- `HKEY_CLASSES_ROOT\Local Settings\Software\Microsoft\Windows\Shell\MuiCache`

5.3 Android Application-Based Forensics

Android is one of the most popular OS used in many smartphones [7, 14]. Android applications often employ hardening techniques to protect themselves; thus, it is essential for forensic examination to perform in-depth analysis [5, 12, 13]. The default location of the MEGA application folder is `/Internal shared storage/Android/data/ mega.privacy.android.app/`.

Installation Artifact By analyzing the forensic image of android activities, we retrieved information related to the MEGA application such as `FirstInstallTime= 20210225T113533Z`, `LastUpdateTime=20210302T065720Z`, Name=MEGA, `Package= mega.privacy.android.app`, `Version= 4.0.1 (355)`, and `AppSize=89248722`, as shown in Fig. 6 [4].



```
0000000 4E61 6D65 3D4D 4547 410D 0A50 6163 6B61 6765 3D6D 6567 612E 7072 6976 6163 792E 616E 6472 6F69 Name=MEGA..Package=mega.privacy.androi
0000026 642E 6170 700D 0A56 6572 7369 6F6E 3D34 2E30 2E31 2028 3335 3529 0D0A 4973 5379 7374 656D 3D30 d.app..Version=4.0.1 (355)..IsSystem=0
000004C 0D0A 4170 7053 697A 653D 3839 3234 3837 3232 0D0A 4461 7461 5369 7A65 3D30 0D0A 4361 6368 6553 ..AppSize=89248722..DataSize=0..CacheS
0000072 697A 653D 300D 0A46 6972 7374 496E 7374 616C 6C54 696D 653D 3230 3231 3032 3235 5431 3133 3533 ize=0..FirstInstallTime=20210225T11353
0000098 335A 0D0A 4C61 7374 5570 6461 7465 5469 6D65 3D32 3032 3130 3330 3254 3036 3537 3230 5A0D 0A49 3Z..LastUpdateTime=20210302T065720Z..I
00000BE 6E73 7461 6C6C 6572 5061 636B 6167 654E 616D 653D 636F 6D2E 616E 6472 6F69 642E 7665 6E64 696E nstallerPackageName=com.android.vendin

0029998 3300 3200 5300 7000 6900 6E00 4300 6F00 7500 6E00 7400 0000 6D00 0000 0000 0000 0EDE 8C22 0007 3.2.S.p.i.n.C.o.u.n.t...m.........."..
00299BE 0080 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 ......................................
00299E4 0000 0000 0BDE 8922 0008 0097 5553 4552 4E41 4D45 3D48 696D 616E 7368 7520 4D69 7368 7261 0000 ......."....USERNAME=Himanshu Mishra..
0029A0A 0000 0000 0000 0000 0000 0000 0000 34DE 8A22 0009 0096 4700 4F00 4200 4A00 4500 4300 5400 5F00 ..............4.."....G.O.B.J.E.C.T._.
0029A30 4400 4500 4200 5500 4700 0000 0000 0000 0000 0000 0000 0000 31DE B722 000A 0080 5000 7200 6F00 D.E.B.U.G...............1.."....P.r.o.
0029A56 7600 6900 6400 6500 7200 4900 6E00 6600 6F00 0000 0000 7000 6100 6300 6500 0000 0000 32DE B022 v.i.d.e.r.I.n.f.o.....p.a.c.e.....2.."
0029A7C 000B 0080 4100 6400 6400 7200 6500 7300 7300 4600 6100 6D00 6900 6C00 7900 0000 0000 0000 D09E ....A.d.d.r.e.s.s.F.a.m.i.l.y.........
0029AA2 6250 6401 0000 3FDE BD22 000C 0095 5072 6F67 7261 6D44 6174 613D 433A 5C50 726F 6772 616D 4461 bPd...?.."....ProgramData=C:\ProgramDa
0029AC8 7461 0000 0000 0000 0000 0000 0000 0000 38DE BE22 000D 0080 5000 7200 6F00 7600 6900 6400 6500 ta..............8.."....P.r.o.v.i.d.e.
0029AEE 7200 4900 6E00 6600 6F00 0000 0000 0000 0000 0000 0000 0000 25DE BB22 000E 0080 4100 6400 r.I.n.f.o...............%.."....A.d.
0029B14 6400 7200 6500 7300 7300 4600 6100 6D00 6900 6C00 7900 0000 0000 0000 0000 0000 0000 0000 26DE d.r.e.s.s.F.a.m.i.l.y...............&.
0029B3A A422 000F 0093 5052 4F43 4553 534F 525F 4152 4348 4954 4543 5455 5245 3D41 4D44 3634 0000 0000 ."....PROCESSOR_ARCHITECTURE=AMD64....
0029B60 0000 0000 0000 0000 23DE A122 0010 0080 5000 7200 6F00 7600 6900 6400 6500 7200 4900 6E00 6600 ........#.."....P.r.o.v.i.d.e.r.I.n.f.
0029B86 6F00 0000 4900 7400 6500 6D00 0000 0000 0000 2CDE A222 0011 0088 0C7B 8254 FC7F 0000 60B8 F751 o...I.t.e.m.........,"......{.T....`..Q
```

Fig. 6 Artifacts located during MEGA Android app forensics

Table 8 Hash value and timestamps of the original and the downloaded files on Android device

Filename	Type	Date/Time	MD5 hash
File7.jpg	Original	2021-02-22 14:26:36	12BFED37A17A0C74FF2DA0D234B47C30
	Downloaded	2021-02-23 19:46:49	12BFED37A17A0C74FF2DA0D234B47C30
File8.png	Original	2021-02-22 14:27:06	D0ACF757945B07B6CFCFF7B8E0D9B0FF
	Downloaded	2021-02-23 19:47:05	D0ACF757945B07B6CFCFF7B8E0D9B0FF

Download Artifact Files downloaded by users are located at `/Internal Shared Storage/MEGA/MEGA Downloads`. Artifacts of uploaded and downloaded files are listed in Table 8.

Memory Artifact In analysis of the memory, a copy of RAM dump forensic was extracted and analyzed using *fridump opensource software*. We were able to locate the username of the account.

Network traffic The communication session between the android application and the MEGA cloud server was established through TLSv1.2 with IP address ipv4.scr 66.203.125.15 and port no 443.

Share Analysis In our investigation, we were able to track down a shared URL `https://mega.nz/file/1U4jFIqA#Zye2CUDArha8swNIrBpiJE1ih-D0iq2ETtyT`
`ZpwssKE`. Accordingly, we investigate the file shared with more than one examiner account. In cases where the client's file-sharing setting is set to the public, the URL of the document connection will show the common file. Regardless, however, the record is kept hidden.

6 Conclusion

In this investigative study, a cloud storage service, such as MEGA, allows clients to store, upload, download, access, and share data 24/7. We described and located various forensic artifacts when the MEGA cloud was used with Google Chrome browser, Windows application, and android application. In this investigation, the recovered artifacts include the name of the created account, email address, network information, username, and check the MD5 hash of the original and downloaded files. In addition, we also studied the metadata of the original and unaltered downloaded files and discovered changes in the timestamp. Apart from these, we were also able to recover the client's credentials. However, we could not locate the user password on both the memory dump and storage. Nevertheless, we could employ offline strategies to achieve that. Lastly, we accomplished a forensic investigation on the client side, in which the artifacts located on the client machine were recovered and further analyzed.

References

1. Choo K-KR, Esposito C, Castiglione A (2017) Evidence and forensics in the cloud: challenges and future research directions. IEEE Cloud Comput 4(3):14–19
2. Darsana M, Khosh VA, Dija S, Indu V (2018) Windows 10 cloud storage and social media application forensics. In: 2018 IEEE international conference on computational intelligence and computing research (ICCIC). IEEE, New York, pp 1–5
3. Daryabar F, Dehghantanha A, Choo K-KR (2017) Cloud storage forensics: mega as a case study. Australian J Forensic Sci 49(3):344–357
4. Grover N, Saxena J, Sihag V (2017) Security analysis of onlinecabbooking android application, pp 603–611
5. Huang C-T, Ko H-J, Zhuang Z-W, Shih P-C, Shiuh-Jeng W (2018) Mobile forensics for cloud storage service on iOS systems. In: 2018 international symposium on information theory and its applications (ISITA). IEEE, New York, pp 178–182
6. Ko H-J, Huang C-T, Zhuang Z-W, Horng G, Wang S-J (2020) Cloud evidence tracks of storage service linking with iOS systems. J Supercomput pp 1–18
7. La Marra A, Martinelli F, Mercaldo F, Saracino A, Sheikhalishahi M (2020) D-bridemaid: a distributed framework for collaborative and dynamic analysis of android malware. J Wireless Mob Networks Ubiquitous Comput Dependable Appl 11(3):1–28
8. Martini B, Choo K-KR (2013) Cloud storage forensics: ownCloud as a case study. Digit Invest 10(4):287–299
9. Quick D, Choo K-KR (2013) Forensic collection of cloud storage data: does the act of collection result in changes to the data or its metadata? Digit Invest 10(3):266–277
10. Satrya GB (2019) Digital forensics study of a cloud storage client: a dropbox artifact analysis. CommIT (Commun Inform Technol) J 13(2):57–66
11. Shariati M, Dehghantanha A, Choo K-KR (2016) SugarSync forensic analysis. Australian J Forensic Sci 48(1):95–117
12. Sihag V, Swami A, Vardhan M, Singh P (2020) Signature based malicious behavior detection in android. In: International conference on computing science, communication and security. Springer, Berlin, pp 251–262
13. Sihag V, Vardhan M, Singh P (2021) A survey of android application and malware hardening. Comput Sci Rev 39:100365
14. Talegaon S, Krishnan R (2020) Administrative models for role based access control in android. J Internet Serv Inf Secur 10(3):31–46

Face Recognition Using VGG16 CNN Architecture for Enhanced Security Surveillance—A Survey

Alashiri Olaitan, Adeyinka Adewale, Sanjay Misra, Akshat Agrawal, Ravin Ahuja, and Jonathan Oluranti

Abstract A review of the web camera surveillance, face recognition, convolution neural network (CNN), digital images are presented in this work. Previous works on face recognition systems for enhanced surveillance-based security are presented together with relevant deep learning concepts and theories relating to convolutional neural networks. In-depth analysis is summarized and presented in concise way.

Keywords VGG-convolutional neural network (VGG-CNN) · Classification · Biometrics · Haar cascade classifiers · Face recognition

1 Introduction

Face recognition (FR) is among the most well-studied aspects of computer vision. Through the use of deep learning algorithms and bigger volume datasets, researchers have subsequently seen substantial development in FR, notably for limited social

A. Olaitan · A. Adewale · J. Oluranti
Center of ICT/ICE Research, Covenant University, Ota, Ogun, Nigeria
e-mail: olaitan.alashiri@covenantuniversity.edu.ng

J. Oluranti
e-mail: jonatha.oluranti@covenantuniversity.edu.ng

S. Misra
Department of Computer Science and Communication, Østfold University College, Halden, Norway
e-mail: sanjay.misra@hiof.no

A. Agrawal (✉)
Amity University, Haryana, India
e-mail: akshatag20@gmail.com

R. Ahuja
Shri Viswakarma Skill University, Gurgaon, Hariyana, India

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*,
Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_80

media web images, such as high-resolution photos of famous faces taken by professional photos [1]. However, the far more difficult FR in unrestrained and low-resolution surveillance imagery, on the other hand, remains unsolved and largely unexplored. In the extent of image analysis besides computer vision, face recognition is a challenging task. Face recognition is a biometric technology that uses a digital image to identify or authenticate a person. It is mostly utilized in security and surveillance. Deep neural networks have lately made great progress in general object recognition [2]. Automatic Face Recognition and Surveillance aid in the development of a secure technology for the upcoming era of computers [3]. The following is indeed the basis for a review of the literature: As stated in [4] variations in time, age, and circumstances have an impact on each person's face, skeletal structure, muscle development, and body composition. Face recognition systems, equally images and videos, are becoming increasingly popular to use. This system is essentially focused on a variety of poses, expressions, and illuminations.

Face recognition has been shown to exhibit compression impacts since the imageries are immediately retained and delivered in a compressed state, while depictions have been tested with extensively, but mostly in uncompressed image files. It addresses challenges like monitoring and image classification and object appearances in a way utilized as a collection and compressed video segments per any blob examining even while working with still-to-videos.

Face recognition was demonstrated in real-time using a camera, an image, or a set of faces tracked in a video by the researchers in [2]. They evaluated the distance among both landmarks and particularly in comparison the test image to various established encoded image landmarks, derived HOG features, and then categorized them notwithstanding the lighting, expression, radiance, aging, transformations (translate, rotate, and scale the image), or pose during the recognition phase. Researchers were able to create an automatic face recognition system using a picture or video of a person's face acquired via a mobile device or a webcam.

Face recognition was achieved by integrating two methods: the histogram of oriented gradient (HOG) and the Convolutional Neural Network (CNN). HOG excels at identifying image edges and corners. When contrasted with the local binary pattern (LBP) [5], which employs all eight dimensions for each pixel, HOG utilizes a specific direction for each pixel. However, the coarseness of the binning used by LBP causes it to lose information. Under complicated changes in light and time conditions, HOG features with fewer dimensions perform better than LBP features. With reduced computing time for feature extraction and a reduced number of feature vector magnitudes, HOG features outperform VGG16, VGG19, and others.

The very same image may reflect different pixel information due to variations in illumination and intensity of the acquired images, which is a significant impediment in identifying a person's face. The acquired image was first processed to grayscale, and afterward, the gradient of each pixel was examined based on the lighter to a darker pixel value in the HOG approach. It spotted all of the faces in an image frame based on the gradient analysis. To set up the posing and projecting an image of the frontal face, a face landmark is being employed.

The CNN learning algorithm was used to detect faces depending on the encoded face of the current frame and already cached encoded faces. Face recognition has long been regarded as a watershed moment in image processing. Even while cameras are now found in almost every home, on the streets, and in businesses, detecting a person from the footage is a time-consuming operation, limiting the security's effectiveness, this is one of the reasons face recognition have to be enhanced for effective use of the webcam for surveillance [6]. They overcame the limitations of using webcams for surveillance by improving the face recognition algorithm. The face recognition algorithm consistently contrasts a live video stream with an uploaded image from the database, so when the specific object that is the person is detected, an incredibly quick alert is sent. Surveillance cameras, particularly those installed at airports and other public locations, maybe an extremely effective tool for locating missing people as well as other wanted individuals. They overcame the detection of more than one face in their work.

Surveillance is essential nowadays as societies depend on them to improve safety and security, especially where crime is likely to occur such as car parks, supermarkets, office environments banks, construction sites, and motorways. Currently, video data is mainly being used for forensic purposes; this makes it lose the benefit of being a pro-active real-time alerting system since most of the crimes are usually discovered after the harm has been done. This leaves room for further research in surveillance that is continuous monitoring to send an alert in real-time. Smart cameras are now being incorporated into intelligent systems for surveillance to recognize looks in a crowd in real-time.

2 Biometrics

Biometrics is body measurements and calculations associated with human features. Physiological appearances are referred to the shape of the body. Examples to mention on which researches are going on few embraces like face recognition, iris recognition, palm veins, Deoxyribonucleic acid (DNA), fingerprint, face recognition, palm, vein, retina, and odor/scent [7–10].

2.1 Face Biometrics

Face recognition finds a useful application in several cases including eliminating duplicate entries in a country's voter registration system preventing a person from registering twice. In access control such as computer logon or office access, security at airports for passengers and airline staff all around the world. It is driver's licensing offices, for next of kin benefit recipients, police bookings, banking, electoral registration, employee IDs, identification of newborns, national identity cards, in surveillance operations, passport verification, criminals list verification at police sector, Visa processing, and Card Security control at ATMs.

While facial recognition can be done reliably, quickly, and continuously in controlled environments, the technology is currently too rigid and general to cope with real-world situations. Aging, transformation in facial hair, viewpoint distinctions, and cluttered contextual are an automatic facial recognition system that faces some significant problems. Face identification is difficult to automate because faces are a type of natural object that does not lend itself to simplistic geometric interpretations. Computer-assisted face recognition has the potential of being able to manage a huge quantity of faces, meanwhile, the human brain has restricted memory [11].

2.2 Face Detection Methods

Detecting Faces Techniques are classified as feature-based techniques, in which characteristics provided by [12] express an individual's identity and image-based techniques. With all its statistics and structural classifier, the feature-based algorithm [13] outlines how local features are obtained and their positions. Similarly, image-based approaches [14] used algebraic processes to define color modifications. The qua-ternion is used to build generalized linear filtering methods and a new color edge detector.

Researchers in [4] classified facial recognition algorithms into two invariant techniques: discriminative and generative approaches. Discriminative approaches rely on basic data such as age, weight, skeletal structure, and body mass to be studied, however generative approaches outline the procedure for feeding the data into the model. The Who Is It database, which was created by developing an effective database, comprises age and weight information as well as facial imagery. The database solely includes public figures in an attempt to show changes in age and weight over time. The program tries to distinguish images that have changed in age and weight over time. The outcome of weight is evaluated besides subsequently, neural networks are taught. Training comes first, followed by testing in a learning-based system. In comparison to other methodologies, the researcher obtained a 28.53% Rank-I identification performance accuracy with a 3.4% minimal error rate. Over decades, the system has attempted to detect an individual's facial appearance, position, aging, actual or artificial, disguise, and plastic surgery as described by certain researchers on covariates of imageries.

Face detection in color photos is challenging once the background is multifaceted and the luminance varies, making skin detection problematic and resulting in false positives. A parallel structure algorithm of skin color recognition was used to enhance detection reliability and to create a classifier using a Gaussian-mixture model and the Ada-boost training algorithm to eliminate false positives. Face Candidates algorithm is used to test the face detection algorithm for the skin color model, and then Ada-boost trained algorithm is used to test the classifier's verification algorithm on several images as training examples. Face recognition has also employed color palettes with various applications such as images retrieval, color palette, and color transfer.

The expression and impression of color amalgamations is conveyed by the mixing of colors classified into abstract categories through a distinctive set of colors. The pattern matching method outlines the entire facial features to associate input and reference patterns for face detection. For a typical human being, the most difficult challenge is to apply the facial recognition retrieval model for a correct match in the shortest amount of time. Exclusively, when relating with non-static or dynamic environments such as live streaming, webcam recording, or viewing real-time video where facial features are not distinct enough to use as an input image. To create such a model, the research presented by [15] created a model for solving both steps, Facial Detection, and Facial Recognition. Pattern recognition in video files is used in the facial detection stage, which is executed using a single picture matching algorithm. The second phase was to deliberate the image input from the camera, which began with a GUI for chopped square frame design to transmit the important key extent for separating facial features from a complicated background. Second, the out-turn picture obtained through the data source is recognized, and the mean is calculated using Successive Mean Quantization Transform (SMQT) and Eigen techniques applied to the images. After that, it breaks up using the Sparse Network of Windows (SNOW) classifier for facial detection at a high-speed rate with no impact on the background context. The method has been tested on 150 input image snapshots collected from a webcam and has been verified to be 100% accurate.

Developed a new Surveillance Face Recognition Challenge, dubbed QMUL-SurvFace, to encourage the development of innovative FR algorithms that are successful and robust for low-resolution surveillance face pictures [16]. The low-resolution facial images were captured from real surveillance videos, not from fake downsampling of high-resolution footage. This baseline contains 463,507 facial images representing 15,573 distinct identities taken in uncooperative surveillance scenarios over a significant period. As a result, QMUL-SurvFace is a true-performance surveillance FR problem with low resolution, motion blur, uncontrolled poses, changing occlusion, poor illumination, and backdrop clutters. Evaluate the FR performances of five sample deep learning face recognition models (DeepID2, CentreFace, Vgg-Face, FaceNet, and SphereFace) against current standards on the QMUL-SurvFace task [16].

Appearance-based Face Detection. To identify the relevant features of the face and non-facial imagery, these methods use statistical analysis and machine learning techniques. The learnt qualities are expressed in the format of distribution models or discriminant functions, which are subsequently used to detect faces. Meanwhile, dimensional minimization is commonly used to increase computation and detection effectiveness.

Feature-based Face Detection. These methods, also referred to as constituent face recognition, rely on the relationship between the components of the face, it employs invariant features of faces for detection. The idea is that humans can detect faces and objects in a variety of positions and lighting environments, hence attributes or features (such as brows, nose, eyes, mouth, and skin color) must be invariant across these variations. A statistical model is initiated based on the retrieved features to depict their relationships and verify the presence of a face. The reliability of visual feature detection is crucial in this approach.

2.3 Face Recognition Methods

Face recognition methods can be grouped broadly into two: Learning-based Methods and Hand-crafted Methods.

Learning-based Methods. The learning-based methods usually employ convolutional neural networks (CNN) of varying configurations and depths (layers). A baseline CNN is made up of several layers, each of which uses a variational function to transfer one volume of activations to another. Its architecture consists of at least a Convolutional Layer, Pooling Layer, and Fully-Connected Layer. Due to their excellent learning ability especially for large-scale input data, they are constantly being employed by more and more researchers [17]. Deep learning's accomplishment in face recognition has recently surpassed those such as handcrafted and machine learning methods [18]. CNN architectures strive to be deeper and much more complex to acquire improved recognition performance, which consumes resources, time, as well as space. Nevertheless, CNN is used to learn and extract useful features from an image, they also have the advantage that different configurations already trained for specific tasks exist and could be adapted. Certain layers of a trained model (typically the last output layer) can be removed, and the activations of the lower levels can then be used as fixed feature extractors. Several studies have achieved promising results using these deep characteristics [19, 20], and [21].

Hand-crafted Methods. The hand-crafted methods are further divided into four broad categories: a global approach, local approach, appearance or holistic approach, and other methods (which do not fall under the first three).

Global Approach. These are features based on the general texture or appearance of the image. There are a lot of global feature extraction approaches in literature but the most widely used are Gabor filters [22]; Histogram of oriented Gradients [23]; Local phase quantization (LPQ) [24]; Discrete Cosine Transform (DCT) [25]; Local Binary Patterns (LBP) [24, 26]; Weber local descriptor (WLD) [23, 27]; Local Oriented Statistics Information Booster (LOSIB) [23].

Local Approach. This approach focuses on the local facial features such as eyes, mouth, and nose, computes their locations, and applies statistical properties, geometry, or appearance as the determining factors for classification. These are traits that are focused on the image's most crucial details and their spatial relationships with each other. The most commonly used textures are Scale Invariant Feature Transformation (SIFT) [28]; Speeded Up Robust Features (SURF) [29]; Symmetry Assessment by Feature Expansion (SAFE) [28]; Binary Robust Invariant Scalable Keypoints

(BRISK) [30]; Oriented FAST and Rotated BRIEF (ORB) [31]; Phase Intensive Local Pattern (PILP) [29].

Holistic Approach. The entire facial region is regarded as data input for the facial capture system in this approach e.g., Eigenfaces, Principal Component Analysis (PCA), Linear Discriminant Analysis and independent component analysis, and so on. The holistic-based technique tries to distinguish a face by employing global representations, that is, the image as a whole. To acquire the feature vectors, methods like Principal Component Analysis (PCA), Independent Component Analysis (ICA), and Linear Discriminant Analysis (LDA) are utilized. Examples of these features include Eigenface (implemented using PCA) and Fisherface (implemented using LDA).

Biologically Inspired Features (BIF). are imitative primate's feed-forward model of visual object recognition pipeline which is acknowledged to be intelligent to recognize visual patterns with high exactness. Gabor functions are employed to model basic cells in mammalian brains' visual cortex. Gabor filters show frequencies and orientations that are similar to frequencies and orientations in the human visual system. As a result, Gabor filter image processing is regarded to be similar to human comprehension in the visual system. These features were applied by [13].

Elastic Bunch Graph Matching (EBGM). EBGM stands for feature-based face recognition. Certain facial traits are selected by manual interaction. These characteristics are used to create a bunch graph. The bunch graph's numerous nodes represent various facial landmarks. We may establish the gap among a given test image trait and the closest accessible train image feature by scanning for the shortest measure and analyzing a single train image to all of the training images. A feature extraction method incorporates both a holistic and a local approach. 3D imagery is used in the majority of hybrid techniques. Also, because the image of a person's face is acquired in 3D, the technology can identify the curves of the eye sockets, including the forms of the chin and forehead. Since the technique uses depth and an axis of assessment, a profile face may be sufficient as it has significant data to build a whole face.

Others. Some researchers [32], have exploited the use of facial marks such as moles and freckles to try to recognize faces though in combination with LBP and Fisher vectors. Other approaches include Active Shape Models (ASM) which uses the shape of an object (face) as its features by a collection of landmark points at clear corners of the face and facial landmark boundaries [33]. AAM builds a shape model and an intensity model from such a collection of training samples using principal component analysis (PCA) [34].

3 Generic Modes of Face Recognition System

A face recognition system comparing it to other biometric recognition systems operates in two modes [35]:

a. The training mode: the face image of an individual is captured using an acquisition sensor like camera and scanner. The acquired face image is processed and stored in the database with a label (name or unique number) for easy identification or verification.
b. The testing mode: the face image stored is once again acquired and processed to obtain the necessary features required to either verify or identify the individual.

3.1 Generic Modules of Face Recognition Systems

A face recognition system as shown in Fig. 1 is designed using the following basic modules. Modules 3 and 4 are carried out with CNN i.e., the CNN architecture is used for the features extraction and the classification stages.

a. Images acquisition: an acquisition sensor like a camera or sensor is used to capture faces from images or videos. The images must have a considerable amount of spatial information about the face before they can be useful.
b. Images pre-processing: the pre-processing entails cropping out faces from the acquired images and performing some enhancement on them, to make subsequent processing easy and also to advance the overall performance of the system.
c. Feature extraction: this involves extracting the low-level features like edges, lines, dots, medium-level features e.g., texture and color, and high-level features e.g., shape from the face images. The features are used for the recognition process.
d. Matching/classification: this compares the features obtained during recognition against the stored images to produce a matching score.
e. Ion against the stored images to engender a matching score.

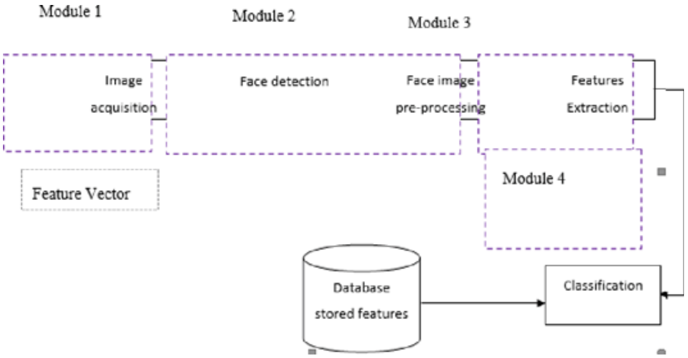


Fig. 1 Block diagram of generic face recognition system

3.2 Overview of Convolutional Neural Network

Convolutional Neural Network (CNN) helps to achieve excellent learning ability for classification of both large-scale and small-scale input data [17]. CNN because of its flexibility and adaptability makes it possible to take out different configurations from already trained models. Convolution simply means applying filters also known as kernels or windows to each image pixel. It tries every possible match. Convolution is performed at each convolutional layer. A layer means a stacking operation and in a convolution layer, the layer consists of the stack images that have been filtered. CNN has been existing since the 1990s but has gained popularity due to its ability to solve recognition problems hence improving computer vision. CNN has its uniqueness from another neural network because of its assumption that all inputs are images, this allows it models its architecture in a way that it recognizes basic image-defined features which help in pattern recognition, face recognition, digits recognition, and many more.

3.3 Overview of Deep Neural Network

Deep learning is a sort of machine learning which enables computers to learn by instance in the likely manner that humans do. Deep learning has progressed to the degree that it can currently beat humans in certain tasks, which include object classification in imagery.

The intrusion detection challenge has been compliant with machine learning methods due to the vast capacity of network telemetry besides other sorts of security data. Numerous modern commercial intrusion detection systems, or security platforms, employ machine learning-based algorithms as part of their detection technique. These methods are often classified as part of the intrusion detection approach's oddity detection class.

There are two types of machine learning models: shallow learning or typical models and deep learning models from 40 machine learning models. Deep learning models are neural network models with a large degree of hidden layers that are currently in use. These models can learn extremely complex nonlinear functions, and hierarchical layering allows them to learn relevant feature representations from incoming data. Deep learning algorithms have recently achieved success in a variety of domains, including image 45 categorization. There are two key reasons deep learning has lately become useful:

a. Deep learning necessitates a significant deal of computational power. A parallel architecture is suited for deep learning on high-performance GPUs. This helps developers to reduce deep learning network training time from weeks to hours each when used during conjunction either clusters or cloud computing.
b. Deep learning requires substantial labeled data. For example, driverless car development requires millions of images and thousands of hours of video. Apart from

scalability, another benefit mention often about deep learning models is their ability to perform automatic feature extraction from raw data, also called feature learning.

Deep learning architectures for example deep neural networks, deep belief networks, convolutional neural networks, and recurrent neural networks have been put into fields including natural language processing, vision speech, computer vision, speech recognition, audio recognition, medical image analysis, machine translation, material inspection, and bioinformatics to mention few where the findings are on par with, if not better than, the efficiency of a human expert. Generally, these architectures can be put into 3 specific categories:

Feed-Forward Neural Networks. This is the least used model of neural networks in practical applications. The first layer is the inputs, while the last layer is the outputs. Neural networks with far more than a hidden layer are referred to as "deep" neural networks. They do a series of calculations that change how related the instances are. Each layer's neurons' activity is a nonlinear function of the previous layer's neurons' activities.

Recurrent Networks. In their connection graph, these have directed cycles. As a result, following the arrows can sometimes lead you back to where you started. These may exhibit complicated dynamics, making training them difficult. They have a physically more realistic aspect to them. There is a great deal of interest right now in figuring out how to train recurrent networks efficiently. Modeling sequential data using recurrent neural networks is a quite natural development. They're similar to very deep nets with one hidden layer each time slice, with the exception that they use the same weights and receive input at each time slice. They possess the ability to recall information for a long time in their concealed condition, but it is extremely difficult to instruct them to use this skill.

Symmetrically Connected Networks. These are similar to recurrent networks, but the unit connections are symmetrical (they have the same weight in both directions). Recurrent networks are substantially more difficult to examine than symmetric networks. As they follow an energy function, they are likewise limited in what they can perform. "Hopfield Nets" are symmetrically linked nets with no hidden units. "Boltzmann machines" are hidden units in an asymmetrically linked network.

4 Related DNN-Based Face Recognition Work

This section reviews existing works related to the development of a face recognition system for enhanced security surveillance. Several research works have been carried out in the field of face recognition from images captured by webcam with impressive results; however, there is still a lot of room for contribution. A summary table is presented in Table 1.

Table 1 Previous work on the development of a face recognition system for enhanced security surveillance

S/N	Authors	Method for features extraction	Method for classification	Database (s) used	Accuracy gotten
1	[36]	Haar-DE filter	MLP, SVM and CNN	Random	N/A
2	[37]	CNN	CNN	VGG	N/A
3	[38]	Viola-Jones and Color-based	Pixel-based skin detection	Random	N/A
4	[39]	CNN	SVM	ORL and FERET	N/A
5	[40]	CNN	CNN	ORL	85.6%
6	[41]	LBP	CNN	VGG16	N/A
7	[11]	B2DPCA, ELM	ELM	FRGC	N/A
8	[35]	Viola-Jones	CNN, PCA	ORL	N/A
9	[42]	Log-ICA	SVM, LDA	CMU, Face Recognition Technology (FERET), YALE	96.45%
10	[43]	Viola-Jones	OFAST, ORB	SVM	N/A
11	[44]	LBP	CNN	VGG-11	83.2%
12	[45]	PCA, BPNN	ELM	FRGC	N/A
13	[46]	DCT	Radial basis function neural network (RBFNN) optimized using firefly algorithm	ORL, Yale, AR, and LFW	97.62%
14	[31]	Gabor analysis	X2 distance	BioSec, CASIA v3, IIT Delhi v1.0, MobBIO, UBIRIS v2 ND-IRIS-0405	N.A
15	[17]	CNN	CNN	Labeled Faces in the Wild (LFW)	94.97
16	[47]	LBPHF, Haar	SVM	Images of Groups (IOG)	90.1%
17	[18]	CNN	CNN	LlFW	98.46%
18	[48]	Singular value decomposition (SVD)	Nearest neighbor (NN) based on Euclidean distance	Yale B, CMU-PIE	97.44%
19	[49]	Phase intensive local pattern (LILP)	Euclidean distance	FERET v4	97.3%
20	[50]	Motif	Euclidean distance, cosine distance and SVM	AR	98%

(continued)

Table 1 (continued)

S/N	Authors	Method for features extraction	Method for classification	Database (s) used	Accuracy gotten
21	[46]	DCT	Radial basis function neural network (RBFNN) optimized using firefly algorithm	ORL, Yale, AR, and LFW	99.83%
22	[51]	CNN	CNN	SIEM bio-medicale	80%
23	[52]	HOG, MMFD	KNN, SVM, DNN	ORL, AR	96.66%
24	[37]	DCT, two directional multi-level threshold-LBP fusion (2D–MTLBP-F)	Sparse representation-based classification (SRC)	FRGC	94.67%
25	[16]	CNN	CNN	LFW	99.11%

5 Summary

After reviewing existing works of literature, Deep Convolutional Neural Network (DCNN) has proved to attain state-of-the-art results for face recognition as a security means to prevent intrusion, 1w especially for large datasets. DCNN also has the advantage over other neural networks for image classification because DCNN automatically detects the important features without any human supervision. This chapter also explains the importance of surveillance concerning face recognition.

Acknowledgements The authors appreciate the sponsorship from Covenant University through its Center for Research, Innovation and Discovery, Covenant University, Ota Nigeria.

References

1. Lumaban MBP, Battung GT (2020) WEBCAM-based surveillance system with face recognition feature. Int J Eng Adv Technol 9
2. Ahamed H, Alam I, Islam MM (2018) HOG-CNN-based real-time face recognition. International conference on advancement in electrical and electronic engineering, pp 1–4
3. Chawla D, Trivedi MC (2018) A comparative study on face detection techniques for security surveillance. A comparative study on face detection techniques for security surveillance, pp 531–541
4. Singh M, Nagpal S, Singh R, Vatsa M (2014) On recognizing face images with weight and age variations. IEEE Access 2:822–830
5. Ghorbani M, Targhi AT, Dehshibi MM (2015) HOG and LBP: towards a robust face recognition system. International conference on digital information management, pp 138–141
6. Kumar PR, Surendar M, Kumar TUMDM (2019) Smart surveillance cam using face recognition algorithm. J Netw Comput Appl

7. Aniche C, Yinka-Banjo C, Ohalete P, Misra S (2021) Biometric e-voting system for cybersecurity. In: Artificial intelligence for cyber security: methods, issues and possible horizons or opportunities. Springer, Cham, pp 105–137
8. Ugot OA, Yinka-Banjo C, Misra S (2021) Biometric fingerprint generation using generative adversarial networks. In: Artificial intelligence for cyber security: methods, issues and possible horizons or opportunities. Springer, Cham, pp 51–83
9. Olanrewaju L, Oyebiyi O, Misra S, Maskeliunas R, Damasevicius R (2020) Secure ear biometrics using circular kernel principal component analysis, Chebyshev transform hashing and Bose–Chaudhuri–Hocquenghem error-correcting codes. SIViP 14(5):847–855
10. Assibong PA, Wogu IAP, Misra S, Makplang D (2020) The utilization of the biometric technology in the 2013 Manyu division legislative and municipal elections in cameroon: an appraisal. In: Advances in electrical and computer technologies. Springer, Singapore, pp 347–360
11. Mohammed AA, Minhas R, Wu QMJ, Sid-Ahmed MA (2011) Human face recognition is based on multidimensional PCA and extreme learning machines. Pattern Recogn 44(10–11):2588–2597. https://doi.org/10.1016/j.patcog.2011.03.013
12. Antón-Rodríguez M, González-Ortega D, Díaz-Pernas F, Martínez-Zarzuela M, Díez-Higuera J (2012) Color-texture image segmentation and recognition through a biologically-inspired architecture. Pattern Recogn Image Anal 22:54–68
13. Choi SE, Lee YJ, Lee SJ, Park KR, Kim J (2011) Age estimation using a hierarchical classifier based on global and local facial features. Pattern Recogn 44(6):1262–1281. https://doi.org/10.1016/j.patcog.2010.12.005
14. Carré P, Denis P, Fernandez-Maloigne C (2014) Spatial color image processing using clifford algebras: application to color active contour. SIViP 8:1357–1372
15. Pattanasethanon P, Savithi C (2012) Human face detection and recognition using web-cam. J Comput Sci 8:1585
16. Mustafah YM, Azman AW, Bigdeli A, Lovell BC (2007) An automated face recognition system for intelligence surveillance: smart camera recognizing faces in the crowd. 2007 1st ACM/IEEE International conference on distributed smart cameras, ICDSC, pp 147–152. https://doi.org/10.1109/ICDSC.2007.4357518
17. Wu X, He R, Sun Z, Tan T (2018) A light CNN for deep face representation with noisy labels. IEEE Trans Inf Forensics Secur 13(11):2884–2896. https://doi.org/10.1109/TIFS.2018.2833032
18. Zheng HH, Zu YX (2018) A normalized light CNN for face recognition. J Phys: conference series 1087(6). https://doi.org/10.1088/1742-6596/1087/6/062015
19. Shang C, Ai H (2018) Cluster convolutional neural networks for facial age estimation. Proceedings—international conference on image processing, ICIP, 2017–Sept, pp 1817–1821. https://doi.org/10.1109/ICIP.2017.8296595
20. Rattani A, Reddy N, Derakhshani R (2018) Convolutional neural network for age classification from smart-phone based ocular images. IEEE international joint conference on biometrics, IJCB 2017, 2018–Jan, pp 756–761. https://doi.org/10.1109/BTAS.2017.8272766
21. Yoo B, Kwak Y, Kim Y, Choi C, Kim J (2018) Multitask learning with weak label expansion. IEEE Signal Proc Lett 25(6):808–812. Retrieved from https://doi.org/10.1109/LSP.2018.2822241
22. Bharadwaj S, Bhatt HS, Vatsa M, Singh R (2010) Periocular biometrics: when iris recognition fails. BTAS, pp 1–6
23. Castrillón-Santana M, Lorenzo-Navarro J, Ramón-Balmaseda E (2016) On using periocular biometric for gender classification in the wild. Pattern Recogn Lett 82:181–189. https://doi.org/10.1016/j.patrec.2015.09.014
24. Xu J, Cha M, Heyman JL, Venugopalan S, Abiantun R, Savvides M (2010) Robust local binary pattern feature sets for periocular biometric identification. IEEE 4th International conference on biometrics: theory, applications and systems, BTAS 2010, pp 3–10. https://doi.org/10.1109/BTAS.2010.5634504

25. Lyle JR, Miller PE, Pundlik SJ, Woodard DL (2012) Soft biometric classification using local appearance periocular region features. Pattern Recogn 45(11):3877–3885. https://doi.org/10.1016/j.patcog.2012.04.027
26. Uzair M, Mahmood A, Mian A, McDonald C (2015) Periocular region-based person identification in visible, infrared, and hyperspectral imagery. Neurocomputing 149:854–867
27. Aginako N, Castrillón-Santana M, Lorenzo-Navarro J, Martínez-Otzeta JM, Sierra B (2017) Periocular and iris local descriptors for identity verification in mobile applications. Pattern Recogn Lett
28. Sequeira AF, Chen L, Ferryman J, Wild P, Alonso-Fernandez F, Bigun J (2017) Cross-spectral iris/periocular recognition competition, in Biometrics. 2017 IEEE international joint conference on, pp 725–732
29. Bakshi S, Sa PK, Majhi B (2015) A novel phase-intensive local pattern for periocular recognition under the visible spectrum. Biocybernetics Biomed Eng 35(1):30–44. https://doi.org/10.1016/j.bbe.2014.05.003
30. Karahan Ş, Karaöz A, Özdemir ÖF, Gü AG, Uludag U (2014) On identification from periocular region utilizing sift and surf. Proceedings-22nd Europeans
31. Alonso-Fernandez F, Bigun J (2016) A survey on periocular biometrics research. Pattern Recogn Lett pp 96–105
32. Uzair B, Menaa F, Khan BA, Mohammad FV, Ahmad VU, Djeribi R, Menaa B (2018) Isolation, purification, structural elucidation, and antimicrobial activities of kocumarin, a novel antibiotic isolated from actinobacterium Kocuria marina CMG S2 associated with the brown seaweed Pelvetiacanaliculata. Microbiol Res 206:186–197. https://doi.org/10.1016/j.micres.2017.10.007
33. Zou F, Li J, Min W (2019) Distributed face recognition based on load balancing and dynamic prediction. Appl Sci (Switzerland) 9(4). https://doi.org/10.3390/app9040794
34. Makhija Y, Sharma RS (2019) Face recognition: novel comparison of various feature extraction techniques, in Harmony search and nature inspired optimization algorithms. Springer, pp 1189–1198
35. Sawhney S, Kacker K, Jain S, Singh N (n.d.) No title. Real-time smart attendance system using face recognition techniques
36. Besnassi M, Neggaz N, Benyettou A (2020) Face detection based on evolutionary Haar filter. Pattern Anal Appl 23(1):309–330
37. Yun W-H et al (2018) Automatic recognition of children engagement from facial video using convolutional neural networks. IEEE Trans Affect Comput 11(4):696–707
38. Tabatabaie ZS et al (2009) A hybrid face detection system using a combination of appearance-based and feature-based methods. Int J Comput Sci Netw Sec 9(5):181–185
39. Wu, Yulin, and Mingyan Jiang (2018) Multi-layer CNN features fusion and classifier optimization for face recognition. Proceedings of the 2018 2nd international conference on computer science and artificial intelligence
40. Aitkenhead MJ, McDonald AJS (2003) A neural network faces a recognition system. Eng Appl Artif Intell 16(3):167–176. https://doi.org/10.1016/S0952-1976(03)00042-3
41. Yang B et al (2017) Facial expression recognition using weighted mixture deep neural network based on double-channel facial images. IEEE Access 6:4630–4640
42. Bhowmik MK et al (2019) Enhancement of robustness of face recognition system through reduced gaussianity in Log-ICA. Expert Syst Appl 116:96–107
43. Sajjad M, Nasir M, Muhammad K, Khan S, Jan Z, Sangaiah AK, Elhoseny M, Baik SW (2020) Raspberry Pi assisted face recognition framework for enhanced law-enforcement services in smart cities. Future Gener Comput Syst 108:995–1007. https://doi.org/10.1016/j.future.2017.11.013
44. Chowdhry DA, Hussain A, Ur Rehman MZ, Ahmad F, Ahmad A, Pervaiz M (2013) Smart security system for the sensitive area using face recognition. Proceedings—2013 IEEE conference on sustainable utilization and development in engineering and technology, IEEE CSUDET 2013, pp 11–14. https://doi.org/10.1109/CSUDET.2013.6670976

45. Chetty G, Sharma D (2006) Distributed face recognition: a multiagent approach. Lecture notes in computer science (Including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics), 4253 LNAI, pp 1168–1175. https://doi.org/10.1007/11893011_148
46. Agarwal V, Bhanot S (2018) Radial basis function neural network-based face recognition using firefly algorithm. Neural Comput Appl 30(8):2643–2660
47. Owandkar M, Kolte A, Peshave D, Jadhav S (2017) Attendance monitoring system using face recognition. Int Res J Eng Technol (IRJET) 4(5):1163–1168. Retrieved from https://www.irjet.net/archives/V4/i5/IRJET-V4I5228.pdf
48. Zhang Y, Hu C, Lu X (2018) Face recognition under varying illumination based on singular value decomposition and retina modeling. Multimedia Tools Appl 77(21):28355–28374
49. Deniz S, Lee D, Kurian G, Altamirano L, Yee D, Ferra M, Hament B, Zhan J, Gewali L, Oh P (2018) Computer vision for attendance and emotion analysis in school settings
50. Olivares-Mercado J et al (2018) Face recognition system based on MOTIF features. J Mod Opt 65(18):2124–2132
51. Trokielewicz M, Szadkowski M (2017) Iris and periocular recognition in Arabian racehorses using deep convolutional neural networks. In: 2017 IEEE international joint conference on biometrics (IJCB). IEEE
52. Gupta SK, Ashwin TS, Reddy Guddeti RM (2018) CVUCAMS: computer vision-based unobtrusive classroom attendance management system. Proceedings—IEEE 18th international conference on advanced learning technologies, ICALT 2018, pp 101–102. https://doi.org/10.1109/ICALT.2018.00131