

Lecture Notes in Networks and Systems 370

Dharma P. Agrawal

Nadia Nedjah

B. B. Gupta

Gregorio Martinez Perez *Editors*

Cyber Security, Privacy and Networking

Proceedings of ICSPN 2021

 Springer

Lecture Notes in Networks and Systems

Volume 370

Series Editor

Janusz Kacprzyk, Systems Research Institute, Polish Academy of Sciences,
Warsaw, Poland

Advisory Editors

Fernando Gomide, Department of Computer Engineering and Automation—DCA,
School of Electrical and Computer Engineering—FEEC, University of Campinas—
UNICAMP, São Paulo, Brazil

Okyay Kaynak, Department of Electrical and Electronic Engineering,
Bogazici University, Istanbul, Turkey

Derong Liu, Department of Electrical and Computer Engineering, University
of Illinois at Chicago, Chicago, USA

Institute of Automation, Chinese Academy of Sciences, Beijing, China

Witold Pedrycz, Department of Electrical and Computer Engineering, University of
Alberta, Alberta, Canada

Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland

Marios M. Polycarpou, Department of Electrical and Computer Engineering,
KIOS Research Center for Intelligent Systems and Networks, University of Cyprus,
Nicosia, Cyprus

Imre J. Rudas, Óbuda University, Budapest, Hungary

Jun Wang, Department of Computer Science, City University of Hong Kong,
Kowloon, Hong Kong

The series “Lecture Notes in Networks and Systems” publishes the latest developments in Networks and Systems—quickly, informally and with high quality. Original research reported in proceedings and post-proceedings represents the core of LNNS.

Volumes published in LNNS embrace all aspects and subfields of, as well as new challenges in, Networks and Systems.

The series contains proceedings and edited volumes in systems and networks, spanning the areas of Cyber-Physical Systems, Autonomous Systems, Sensor Networks, Control Systems, Energy Systems, Automotive Systems, Biological Systems, Vehicular Networking and Connected Vehicles, Aerospace Systems, Automation, Manufacturing, Smart Grids, Nonlinear Systems, Power Systems, Robotics, Social Systems, Economic Systems and other. Of particular value to both the contributors and the readership are the short publication timeframe and the world-wide distribution and exposure which enable both a wide and rapid dissemination of research output.

The series covers the theory, applications, and perspectives on the state of the art and future developments relevant to systems and networks, decision making, control, complex processes and related areas, as embedded in the fields of interdisciplinary and applied sciences, engineering, computer science, physics, economics, social, and life sciences, as well as the paradigms and methodologies behind them.

Indexed by SCOPUS, INSPEC, WTI Frankfurt eG, zbMATH, SCImago.

All books published in the series are submitted for consideration in Web of Science.

For proposals from Asia please contact Aninda Bose (aninda.bose@springer.com).

More information about this series at <https://link.springer.com/bookseries/15179>

Dharma P. Agrawal · Nadia Nedjah · B. B. Gupta ·
Gregorio Martinez Perez
Editors

Cyber Security, Privacy and Networking

Proceedings of ICSPN 2021

 Springer

Editors

Dharma P. Agrawal
University of Cincinnati
Cincinnati, OH, USA

Nadia Nedjah
State University of Rio de Janeiro
Rio de Janeiro, Brazil

B. B. Gupta
Department of Computer Science
and Information Engineering
Asia University
Taichung, Taiwan

Gregorio Martinez Perez 
University of Murcia
Murcia, Spain

ISSN 2367-3370

ISSN 2367-3389 (electronic)

Lecture Notes in Networks and Systems

ISBN 978-981-16-8663-4

ISBN 978-981-16-8664-1 (eBook)

<https://doi.org/10.1007/978-981-16-8664-1>

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022

This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

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

Organization

Organizing Committee

Honorary Chairs

Jie Wu, Temple University, USA
Valentina E. Balas, Aure IVlaicu University of Arad, Romania
Amiya Nayak, Professor, University of Ottawa, Canada
Michael Sheng, Macquarie University, Sydney, Australia

General Chairs

Dharma P. Agrawal, University of Cincinnati, USA
Nadia Nedjah, State University of Rio de Janeiro, Brazil
Gregorio Martinez Perez, University of Murcia (UMU), Spain

Program Chairs

Kuan-Ching Li, Providence University, Taiwan
B. B. Gupta, Asia University, Taiwan
Francesco Palmieri, University of Salerno, Italy

Publicity Chairs

Anagha Jamthe, University of Texas, Austin, USA
Ahmed A. Abd El-Latif, Menoufia University, Egypt
Nalin A.G. Arachchilage, La Trobe University, Australia

Publication Chairs

Deepak Gupta, Founder and CEO, LoginRadius Inc., Canada
Shingo Yamaguchi, Yamaguchi University, Japan
Nalin A. G. Arachchilage, La Trobe University, Australia

Industry Chairs

Srivathsan Srinivasagopalan, AT T, USA

Suresh Veluru, United Technologies Research Centre Ireland, Ltd., Ireland

Sugam Sharma, Founder CEO, eFeed-Hungers.com, USA

Preface

The International Conference on Cyber Security, Privacy and Networking (ICSPN 2021), held online, is a forum intended to bring high-quality researchers, practitioners, and students from a variety of fields encompassing interests in massive scale complex data networks and big data emanating from such networks.

Core topics of interest included security and privacy, authentication, privacy and security models, intelligent data analysis for security, big data intelligence in security and privacy, deep learning in security and privacy, identity and trust management, AI and machine learning for security, data mining for security and privacy, data privacy, etc. The conference welcomes papers of either practical or theoretical nature, presenting research or applications addressing all aspects of security, privacy, and networking, that concerns to organizations and individuals, thus creating new research opportunities. Moreover, the conference program will include various tracks, special sessions, invited talks, presentations delivered by researchers from the international community, and keynote speeches. A total of 98 papers were submitted, from which 34 were accepted as regular papers.

This conference would not have been possible without the support of a large number of individuals. First, we sincerely thank all authors for submitting their high-quality work to the conference. We also thank all technical program committee members and reviewers and sub-reviewers for their willingness to provide timely and detailed reviews of all submissions. Working during the COVID-19 pandemic was especially challenging, and the importance of team work was all the more visible as we worked toward the success of the conference. We also offer our special thanks to the publicity and publication chairs for their dedication in disseminating the call, and encouraging participation in such challenging times, and the preparation of these proceedings. Special thanks are also due to the special tracks chair, finance chair, and

the web chair. Lastly, the support and patience of Springer staff members throughout the process are also acknowledged.

Cincinnati, USA
Rio de Janeiro, Brazil
Taichung, Taiwan
Murcia, Spain
October 2021

Dharma P. Agrawal
Nadia Nedjah
B. B. Gupta
Gregorio Martinez Perez

Contents

A New Modified MD5-224 Bits Hash Function and an Efficient Message Authentication Code Based on Quasigroups	1
Umesh Kumar and V. Ch. Venkaiah	
Leveraging Transfer Learning for Effective Recognition of Emotions from Images: A Review	13
Devangi Purkayastha and D. Malathi	
An Automated System for Facial Mask Detection and Face Recognition During COVID-19 Pandemic	25
Swati Shinde, Pragati Janjal, Gauri Pawar, Rutuja Rashinkar, and Swapnil Rokade	
ROS Simulation-Based Autonomous Navigation Systems and Object Detection	37
Swati Shinde, Tanvi Mahajan, Suyash Khachane, Saurabh Kulkarni, and Prasad Borle	
Robotic Assistant for Medicine and Food Delivery in Healthcare	49
Akash Bagade, Aditya Kulkarni, Prachi Nangare, Prajakta Shinde, and Santwana Gudadhe	
Privacy-Preserving Record Linkage with Block-Chains	61
Apoorva Jain and Nisheeth Srivastava	
Performance Analysis of Rectangular QAM Schemes Over Various Fading Channels	71
Siddhant Bhatnagar, Shivangi Shah, and Rachna Sharma	
New Symmetric Key Cipher Based on Quasigroup	83
Umesh Kumar, Aayush Agarwal, and V. Ch. Venkaiah	
Validate Merchant Server for Secure Payment Using Key Distribution	95
A. Saranya and R. Naresh	

Extractive Text Summarization Using Feature-Based Unsupervised RBM Method	105
Grishma Sharma, Subhashini Gupta, and Deepak Sharma	
Depression and Suicide Prediction Using Natural Language Processing and Machine Learning	117
Harnain Kour and Manoj Kumar Gupta	
Automatic Detection of Diabetic Retinopathy on the Edge	129
Zahid Maqsood and Manoj Kumar Gupta	
A Survey on IoT Security: Security Threads and Analysis of Botnet Attacks Over IoT and Avoidance	141
M. Vijayakumar and T. S. Shiny Angel	
A Coherent Approach to Analyze Sentiment of Cryptocurrency	155
Ayush Hans, Kunal Ravindra Mohadikar, and Ekansh	
Supervised Machine Learning Algorithms Based on Classification for Detection of Distributed Denial of Service Attacks in SDN-Enabled Cloud Computing	165
Anupama Mishra and Neena Gupta	
Edge Computing-Based DDoS Attack Detection for Intelligent Transportation Systems	175
Akshat Gaurav, B. B. Gupta, and Kwok Tai Chui	
An Empirical Study of Secure and Complex Variants of RSA Scheme	185
Raza Imam and Faisal Anwer	
Text Normalization Through Neural Models in Generating Text Summary for Various Speech Synthesis Applications	197
P. N. K. Varalakshmi and Jagadish S. Kallimani	
Classification of Network Intrusion Detection System Using Deep Learning	207
Neha Sharma and Narendra Singh Yadav	
Toward Big Data Various Challenges and Trending Applications	219
Bina Kotiyal and Heman Pathak	
Convolutional Neural Network-Based Approach to Detect COVID-19 from Chest X-Ray Images	231
P. Pandiaraja and K. Muthumanickam	
Classification of Medical Health Records Using Convolutional Neural Networks for Optimal Diagnosis	247
M. H. Chaithra and S. Vagdevi	

Smart Farming Using IoT Sensors 259
J. Y. Srikrishna and J. Sangeetha

Securing the Smart Devices in Home Automation System 273
Syeda Sabah Sultana and J. Sangeetha

**Dual-Channel Convolutional Recurrent Networks
for Session-Based Recommendation** 287
Jingjing Wang, Lap-Kei Lee, and Nga-In Wu

**Reuse Your Old Smartphone: Automatic Surveillance Camera
Application** 297
Lap-Kei Lee, Ringo Pok-Man Leung, and Nga-In Wu

A Model of UAV-Based Waste Monitoring System for Urban Areas 309
Dalibor Dobrilovic, Gordana Jotanovic, Aleksandar Stjepanovic,
Goran Jausevac, and Dragan Perakovic

**A Secure Multicontroller SDN Blockchain Model for IoT
Infrastructure** 321
K. Janani and S. Ramamoorthy

A Recent Survey on Cybercrime and Its Defensive Mechanism 339
Garima Bajaj, Saurabh Tailwal, and Anupama Mishra

**A Hybrid Feature Selection Approach-Based Android Malware
Detection Framework Using Machine Learning Techniques** 347
Santosh K. Smmarwar, Govind P. Gupta, and Sanjay Kumar

**Security of Big Data: Threats and Different Approaches Towards
Big Data Security** 357
Yashi Chaudhary and Heman Pathak

Segmentation of Image Using Hybrid K-means Algorithm 369
Roopa Kumari and Neena Gupta

A Chatbot for Promoting Cybersecurity Awareness 379
Yin-Chun Fung and Lap-Kei Lee

**An Advanced Irrigation System Using Cloud-Based IoT Platform
ThingSpeak** 389
Salman Ashraf and A. Chowdhury

Author Index 399

Editors and Contributors

About the Editors

Dharma P. Agrawal (M'77-F'87-LF'12) received the B.E. degree in electrical engineering from the National Institute of Technology Raipur, India, in 1966, the M.E. (Honors) degree in electronics and communication engineering from the IIT Roorkee, India, in 1968, and the D.Sc. degree in electrical engineering from EPFL Lausanne, Switzerland, in 1975. He is the Ohio Board of Regents Distinguished Professor at University of Cincinnati, Ohio. His recent research interests include resource allocation and security in mesh networks, efficient deployment and security in sensor networks, use of Femto cells in numerous applications, efficient resource selection in heterogeneous wireless networks, vehicular area networks and use of sensors in monitoring human health and fitness of athletes. His recent contribution in the form of a co-authored introductory text book on wireless and mobile computing has been widely accepted throughout the world, and a third edition has been published. The book has been reprinted both in China and India and translated into Korean and Chinese languages. His co-authored book on Ad hoc and Sensor Networks, second edition published in spring of 2011, is called the best seller by the publisher. He has delivered keynote speeches at 26 different international conferences. He has published 625 papers, given 42 different tutorials and extensive training courses in various conferences in the USA, and numerous institutions in Taiwan, Korea, Jordan, UAE, Malaysia and India in the areas of Ad hoc and Sensor Networks and Mesh Networks, including security issues. He has been appointed as the founding Editor-in-Chief of the *Central European Journal of Computer Science*, Versita. He has graduated 64 Ph.D. and 55 M.S. students. He has also been named as an ISI Highly Cited Researcher in Computer Science. He is a recipient of 2008 Harry Goode Memorial award from the IEEE Computer Society, 2011 Award for Excellence in Mentoring of Doctoral Students, University of Cincinnati, and founding Fellow of the National Academy of Inventors, 2012. He is a Life Fellow of the IEEE.

Nadia Nedjah received the engineering degree in computer science and the M.Sc. degree in system engineering and computation from the University of Annaba, Algeria, and the Ph.D. degree in computation from the University of Manchester Institute of Science and Technology, Manchester, UK. She is Associate Professor with the Department of Electronics Engineering and Telecommunications, Faculty of Engineering, State University of Rio de Janeiro, Brazil. Her research interests include functional programming, embedded systems and reconfigurable hardware design, as well as cryptography.

B. B. Gupta received the Ph.D. degree in information and cyber security from IIT Roorkee, India. He has published more than 400 research articles in international journals and conferences of high repute, including the IEEE, Elsevier, ACM, Springer and Inderscience. He is Professor with Department of Computer Science and Information Engineering, Asia University, Taichung 413, Taiwan. He is also a senior Member of IEEE, ACM, and Life Member of the International Association of Engineers (IAENG) and the International Association of Computer Science and Information Technology (IACSIT). He is selected as a distinguished lecturer for IEEE Consumer Technology Society and also included in the list of Top 2% Scientists in the world from Stanford University USA. He also received the Sir Visvesvaraya Young Faculty Research Fellowship Award, in 2017, from the Ministry of Electronics and Information Technology, Government of India. He also received the 2019 and 2018 Best Faculty Award for Research Activities from NIT Kurukshetra. He has been serving/served as Associate Editor for the *IEEE TII*, *IEEE ITS*, *ACM TOIT*, *IEEE Access*, *IEEE IoT*, etc. He is also leading the *International Journal of Cloud Applications and Computing* (IJCAC), as Editor-in-Chief. His research interests include information security, cyber security, mobile/smartphone, cloud computing, web security, intrusion detection, computer networks and phishing.

Gregorio Martínez Perez is currently Full Professor with the University of Murcia, Murcia, Spain, since 2014. His scientific activity is mainly devoted to cyber security and data science. He is working on different national and European IST research projects related to these topics, being principal investigator for UMU in most of them. He received the Ph.D. degree in computer science with the University of Murcia.

Contributors

Aayush Agarwal Manipal Institute of Technology, Manipal, India

Faisal Anwer Department of Computer Science, Aligarh Muslim University, Aligarh, India

Salman Ashraf Electronics and Communication Engineering Department, NIT Agartala, Agartala, India

Akash Bagade Pimpri Chinchwad College of Engineering, Pune, India

Garima Bajaj Swami Rama Himalayan University, Dehradun, India

Siddhant Bhatnagar Institute of Technology, Nirma University, Ahmedabad, GJ, India

Prasad Borle Pimpri Chinchwad College of Engineering, Pune, Maharashtra, India

M. H. Chaithra Department of Computer Science and Engineering, REVA University, Bangalore, India;

Visvesvaraya Technological University, Belagavi, Karnataka, India

Yashi Chaudhary Gurukul Kangri University, Haridwar, India

A. Chowdhury Electronics and Communication Engineering Department, NIT Agartala, Agartala, India

Kwok Tai Chui School of Science and Technology, Hong Kong Metropolitan University, Clear Water Bay, Hong Kong, China

Dalibor Dobrilovic Technical Faculty “Mihajlo Pupin” Zrenjanin, University of Novi Sad, Zrenjanin, Serbia

Ekansh National Institute of Technology Kurukshetra, Kurukshetra, Haryana, India

Yin-Chun Fung School of Science and Technology, Hong Kong Metropolitan University, Ho Man Tin, Kowloon, Hong Kong SAR, China

Akshat Gaurav Ronin Institute, Montclair, NJ, USA

Santwana Gudadhe Pimpri Chinchwad College of Engineering, Pune, India

B. B. Gupta Department of Computer Science and Information Engineering, Asia University, Taichung, Taiwan

Govind P. Gupta Department of Information Technology, National Institute of Technology Raipur, Raipur, India

Manoj Kumar Gupta School of Computer Science and Engineering, Shri Mata Vaishno Devi University, Katra, Jammu & Kashmir, India

Neena Gupta Department of Computer Science, Gurukul Kangri Deemed to University, Kanya Gurukul Campus, Dehradun, Haridwar, Uttarakhand, India

Subhashini Gupta Department of Computer Science, K. J. Somaiya College of Engineering, Mumbai, India

Ayush Hans National Institute of Technology Kurukshetra, Kurukshetra, Haryana, India

Raza Imam Department of Computer Science, Aligarh Muslim University, Aligarh, India

Apoorva Jain Department of CSE, IIT Kanpur, Kanpur, India

K. Janani Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur, India

Pragati Janjal Computer Engineering Department, Pimpri Chinchwad College of Engineering, Pune, India

Goran Jausevac Faculty of Transport and Traffic Engineering, University of East Sarajevo, Dobo, Bosnia and Herzegovina

Gordana Jotanovic Faculty of Transport and Traffic Engineering, University of East Sarajevo, Dobo, Bosnia and Herzegovina

Jagadish S. Kallimani Professor and Head, Department of Artificial Intelligence & Machine Learning, M S Ramaiah Institute of Technology, Bangalore, India

Suyash Khachane Pimpri Chinchwad College of Engineering, Pune, Maharashtra, India

Bina Kotiyal Gurukul Kangri Vishwavidyalaya, Dehradun, Uttarakhand, India

Harnain Kour School of Computer Science and Engineering, SMVDU, Katra, Jammu & Kashmir, India

Aditya Kulkarni Pimpri Chinchwad College of Engineering, Pune, India

Saurabh Kulkarni Pimpri Chinchwad College of Engineering, Pune, Maharashtra, India

Sanjay Kumar Department of Information Technology, National Institute of Technology Raipur, Raipur, India

Umesh Kumar School of Computer & Information Sciences, University of Hyderabad, Hyderabad, India

Roopa Kumari Department of Computer Science, Gurukul Kangri Deemed to University, Kanya Gurukul Campus, Dehradun, Haridwar, Uttarakhand, India

Lap-Kei Lee School of Science and Technology, Hong Kong Metropolitan University, Ho Man Tin, Kowloon, Hong Kong SAR, China

Ringo Pok-Man Leung The Executive Centre Limited, Central, Hong Kong SAR, China

Tanvi Mahajan Pimpri Chinchwad College of Engineering, Pune, Maharashtra, India

D. Malathi Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur, India

Zahid Maqsood Shri Mata Vaishno Devi University, Katra, J&K, India

Anupama Mishra Gurukul Kangri Vishwavidyalaya, Haridwar, India; Swami Rama Himalayan University, Dehradun, India

Kunal Ravindra Mohadikar National Institute of Technology Kurukshetra, Kurukshetra, Haryana, India

K. Muthumanickam Kongunadu College of Engineering and Technology, Thottiam, Tiruchirappalli, Tamil Nadu, India

Prachi Nangare Pimpri Chinchwad College of Engineering, Pune, India

R. Naresh Department of Computer Science and Engineering, SRM Institute of Science and Technology, Chennai, Tamil Nadu, India

P. Pandiaraja Department of Computer Science and Engineering, M.Kumarasamy College of Engineering, Karur, Tamil Nadu, India

Heman Pathak Gurukul Kangri Vishwavidyalaya, Dehradun, Uttarakhand, India; Gurukul Kangri University, Haridwar, India

Gauri Pawar Computer Engineering Department, Pimpri Chinchwad College of Engineering, Pune, India

Dragan Perakovic Faculty of Transport and Traffic Sciences, University of Zagreb, Zagreb, Croatia

Devangi Purkayastha Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur, India

S. Ramamoorthy Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur, India

Rutuja Rashinkar Computer Engineering Department, Pimpri Chinchwad College of Engineering, Pune, India

Swapnil Rokade Computer Engineering Department, Pimpri Chinchwad College of Engineering, Pune, India

J. Sangeetha Department of Computer Science and Engineering, M S Ramaiah Institute of Technology, Bangalore, India

A. Saranya Department of Computer Science and Engineering, SRM Institute of Science and Technology, Chennai, Tamil Nadu, India

Shivangi Shah Institute of Technology, Nirma University, Ahmedabad, GJ, India

Deepak Sharma Department of Computer Science, K. J. Somaiya College of Engineering, Mumbai, India

Grishma Sharma Department of Computer Science, K. J. Somaiya College of Engineering, Mumbai, India

Neha Sharma Manipal University Jaipur, Jaipur, Rajasthan, India

Rachna Sharma Institute of Technology, Nirma University, Ahmedabad, GJ, India

Prajakta Shinde Pimpri Chinchwad College of Engineering, Pune, India

Swati Shinde Computer Engineering Department, Pimpri Chinchwad College of Engineering, Pune, India

T. S. Shiny Angel SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India

Santosh K. Smmarwar Department of Information Technology, National Institute of Technology Raipur, Raipur, India

J. Y. Srikrishna Department of Computer Science and Engineering, M S Ramaiah Institute of Technology, Bangalore, India

Nisheeth Srivastava Department of CSE, IIT Kanpur, Kanpur, India

Aleksandar Stjepanovic Faculty of Transport and Traffic Engineering, University of East Sarajevo, Dobo, Bosnia and Herzegovina

Syeda Sabah Sultana Department of Computer Science and Engineering, M S Ramaiah Institute of Technology, Bengaluru, India

Saurabh Tailwal Swami Rama Himalayan University, Dehradun, India

S. Vagdevi Department of Computer Science and Engineering, City Engineering College, Bangalore, India

P. N. K. Varalakshmi Research Scholar, Department of Computer Science and Engineering, M S Ramaiah Institute of Technology, Bangalore, India

V. Ch. Venkaiah School of Computer & Information Sciences, University of Hyderabad, Hyderabad, India

M. Vijayakumar SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India

Jingjing Wang School of Science and Technology, Hong Kong Metropolitan University, Ho Man Tin, Hong Kong SAR, China

Nga-In Wu College of Professional and Continuing Education, Hong Kong Polytechnic University, Kowloon, Hong Kong SAR, China

Narendra Singh Yadav Manipal University Jaipur, Jaipur, Rajasthan, India

A New Modified MD5-224 Bits Hash Function and an Efficient Message Authentication Code Based on Quasigroups



Umesh Kumar and V. Ch. Venkaiah

Abstract In this paper, we have proposed (i) a hash function and (ii) an efficient message authentication code based on quasigroup. We refer to these as QGMD5 and QGMAC, respectively. The proposed new hash function QGMD5 is an extended version of MD5 that uses an optimal quasigroup along with two operations named as QGExp and QGComp. The operations quasigroup expansion (QGExp) and the quasigroup compression (QGComp) are also defined in this paper. QGMAC is designed using the proposed hash function QGMD5 and a quasigroup of order 256 as the secret key. The security of QGMD5 is analyzed by comparing it with both the MD5 and the SHA-244. It is found that the proposed QGMD5 hash function is more secure. Also, QGMAC is analyzed against the brute-force attack. It is resistant to this attack because of the exponential number of quasigroups of its order. It is also analyzed for the forgery attack, and it is found to be resistant. In addition, we compared the performance of the proposed hash function to that of the existing MD5 and SHA-224. Similarly, the performance of the proposed QGMAC is compared with that of the existing HMAC-MD5 and HMAC-SHA-224. The results show that the proposed QGMD5 would take around $2 \mu s$ additional execution time from that of MD5 but not more than SHA-224, while QGMAC always takes less time than that of both the HMAC-MD5 and the HMAC-SHA-224. So, our schemes can be deployed in all the applications of hash functions, such as in blockchain and for verifying the integrity of messages.

Keywords Cryptography · HMAC-MD5 · HMAC-SHA-224 · Latin square · MD5 · QGMAC · QGMD5 · Quasigroup · SHA-224

U. Kumar (✉) · V. Ch. Venkaiah
School of Computer & Information Sciences, University of Hyderabad, Hyderabad, India
e-mail: kumar.umesh285@gmail.com

V. Ch. Venkaiah
e-mail: vvcs@uohyd.ernet.in

1 Introduction

These days the need for securing a message has been increasing and with that there has been a tremendous need for new hashing techniques and message authentication codes. In cryptography, two types of hash functions are used: (1) hash function without a key (or simply a hash function) and (2) hash function with a key (or HMAC)

1.1 Hash Function Without a Key

A hash function takes an arbitrary length input message and produces a fixed length hash value, called the message digest or checksum. It detects the integrity of a message which is sent by a sender. The properties of the cryptographic hash function (H) are given in [12, 14]

Various cryptographic hash functions exist in the literature [3, 8]. Of these, MD5 is still a widely used hash function because it is one of the hash functions requiring the least number of operations. Of late, many articles are published showing that the MD5 is not secure because the length of the hash-value is too short. So, it is vulnerable to brute force birthday attacks [15], and a collision can be found within seconds with a complexity of around 2^{24} [18]. It is also vulnerable to pre-image attacks and can be cryptanalyzed using dictionary and rainbow table attacks [5, 19]. Various researchers have analyzed the MD5 algorithm against these attacks and tried to modify it [2, 11]. However, no amendment has yet been proven to be fully effective at resolving the vulnerability and therefore remains a challenge to address the problem against MD5 attacks.

1.2 Hash Function with Key or HMAC

The output of HMAC is used to verify both the authenticity and the data integrity of a message when two authorized parties communicate in an insecure channel. It is also used in Internet security protocols, including SSL/TLS, SSH, IPsec. HMAC uses a hash function (H) and a secret key (k) shared between the sender and the receiver. The properties of the HMAC are given in [12, 14].

The security of the proposed schemes is studied by verifying the basic properties of hash function and message authentication code. It is heartening to note that the schemes not just meet the requirements but rather surpass them. Initially, our schemes start with an optimal quasigroup of order 16. Later on, we would like to use optimal quasigroups of order 256.

The paper is organized as follows: Next section gives a brief overview of quasigroup, optimal quasigroup, and MD5. The proposed algorithm including the QGExp

and QGComp operations is discussed in Sect. 3. The performance of the QGMD5 and QGMAC algorithms and its comparison with that of MD5, SHA-224, HMAC-MD5, and HMAC-SHA-224 are discussed in Sect. 4. The security analysis of the proposed QGMD5 and QGMAC is discussed in Sect. 5. The concluding remark is given in Sect. 6.

2 Preliminaries

2.1 Quasigroup

Definition-1: A quasigroup $Q=(Z_n, *)$ is a finite nonempty set Z_n of non-negative integers along with a binary operation '*', satisfying the following properties:

- (i) If $x, y \in Z_n$ then $x*y \in Z_n$ (Closure property).
- (ii) For $\forall x, y \in Z_n, \exists$ unique $a, b \in Z_n$ such that $x * a = y$ and $b * x = y$.

Example 1: Table 1 is an example of a quasigroup of order 3 over the set $Z_3=\{0,1,2\}$. Note that for $x = 2$ and $y = 1, a = 0$ and $b = 1$ are the unique elements of Z_3 such that $x * a = y$ and $b * x = y$, where $*$ denotes the quasigroup operation of order 3. It is true for all $x, y \in Z_3$.

Observe that in a quasigroup, every element appears exactly once in each row and once in each column. Such a table is also called a Latin square [1]. So, the number of quasigroups is the same as that of the Latin squares and the number of quasigroups increases rapidly with its order [17]. In fact, the number is given by the following inequality [9].

$$\prod_{\ell=1}^n (\ell!)^{\frac{n}{\ell}} \geq QG(n) \geq \frac{(n!)^{2n}}{n^{n^2}}, \tag{1}$$

where $QG(n)$ denotes the number of quasigroups of order n . For $n = 2^k, k = 4, 8$ the bounds of the number are:

$$0.689 \times 10^{138} \geq QG(16) \geq 0.101 \times 10^{119}, \tag{2}$$

$$0.753 \times 10^{102805} \geq QG(256) \geq 0.304 \times 10^{101724}. \tag{3}$$

Table 1 Quasigroup of order 3

*	0	1	2
0	2	1	0
1	0	2	1
2	1	0	2

Table 2 Optimal Quasigroup of order 16

*2	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	8	0	11	2	9	14	7	6	13	3	15	4	5	10	12	1
1	11	2	8	0	7	6	9	14	15	4	13	3	12	1	5	10
2	2	11	0	8	6	7	14	9	4	15	3	13	1	12	10	5
3	10	5	1	12	3	13	4	15	14	9	6	7	0	8	2	11
4	9	14	7	6	8	0	11	2	5	10	12	1	13	3	15	4
5	0	8	2	11	14	9	6	7	3	13	4	15	10	5	1	12
6	12	1	5	10	15	4	13	3	7	6	9	14	11	2	8	0
7	1	12	10	5	4	15	3	13	6	7	14	9	2	11	0	8
8	14	9	6	7	0	8	2	11	10	5	1	12	3	13	4	17
9	7	6	9	14	11	2	8	0	12	1	5	10	15	4	13	3
10	3	13	4	15	10	5	1	12	0	8	2	11	14	9	6	7
11	6	7	14	9	2	11	0	8	1	12	10	5	4	15	3	13
12	5	10	12	1	13	3	15	4	9	14	7	6	8	0	11	2
13	4	15	3	13	1	12	10	5	2	11	0	8	6	7	14	9
14	15	4	13	3	12	1	5	10	11	2	8	0	7	6	9	14
15	13	3	15	4	5	10	12	1	8	0	11	2	9	14	7	6

2.2 Optimal Quasigroups

A quasigroup of order 2^k that consists of a collection of $k \times k$ bits optimal S-boxes is called an optimal quasigroup. Our hash function (QGMD5) uses 4×4 bits S-boxes to form an optimal quasigroup. The description of a 4×4 bits optimal S-box is given in [10]. Various approaches to generate the optimal S-boxes of 4×4 bits are given in [10, 13]. Not all such S-boxes are capable of forming the quasigroups because quasigroup is a mathematical object and has certain properties to be satisfied. We have used 16 S-boxes that are suitable for forming the quasigroup, and these are listed row-wise in Table 2.

2.3 Brief Description of MD5

MD5 is the most widely used hash function in cryptography. It is designed based on Merkle–Damgård construction. It takes variable length input (message M) and produces a fixed length 128-bit output (hash-value). Before starting the process, the whole message is divided into 512-bit fixed size blocks. If a message length is not a multiple of 512 bits, then it is padded as given in [16].

Now each 512 bits message block m is divided into sixteen 32-bit words (16 sub-blocks) as $m = m_0, m_1, \dots, m_{15}$. The algorithm of MD5 has four rounds, and each round has 16 steps making 64 steps in total. These four round functions are defined by the following four nonlinear Boolean functions:

$$\begin{aligned}
 R_{(1,j)}(x, y, z) &= (x \wedge y) \vee (\neg x \wedge z), & 1 \leq j \leq 16 \\
 R_{(2,j)}(x, y, z) &= (x \wedge z) \vee (y \wedge \neg z), & 17 \leq j \leq 32 \\
 R_{(3,j)}(x, y, z) &= (x \oplus y \oplus z), & 33 \leq j \leq 48 \\
 R_{(4,j)}(x, y, z) &= y \oplus (x \vee \neg z), & 49 \leq j \leq 64
 \end{aligned} \tag{4}$$

where x, y, z are 32-bit words and \wedge, \vee, \oplus , and \neg are AND, OR, XOR, and NOT operations, respectively. The $R_{(r,j)}$ is defined as the j^{th} step of round r , $1 \leq r \leq 4$ and $1 \leq j \leq 64$.

3 Proposed Schemes

In this section, we have proposed two schemes based on quasigroup: (i) a new hash function QGMD5: it expands the hash size of MD5 and converts 128 bits into 224 bits and (ii) a new message authentication code named here as QGMAC, which is based on the QGMD5. It expands the MD5-based message authentication code (MAC-MD5) to 224 bits. Both the expansions are done through a series of QGExp and QGComp operations. The underlying structure of both the QGMD5 and the QGMAC is similar. The only difference between the two is that the quasigroup used in QGMD5 is publicly known, while the quasigroup used in QGMAC is a secret key. Figure 1 depicts the workflow of both the QGMD5 and the QGMAC. In these schemes, at first, an arbitrary length message is divided into k fixed size blocks, each of which is 512 bits in size. If the length of a message is not a multiple of 512 bits, then the padding will be required, and it is padded as in the case of MD5 hash function [16]. Observe that each round, except the last round of the last block of MD5, is followed by a QGExp operation that inserts 96 bits and a QGComp operation that deletes 96 bits. The last round of the last block of MD5 is followed by only a QGExp operation. QGExp and QGComp are denoted by \otimes and \odot , respectively. Since our proposed schemes use quasigroups of orders 16 and 256, the functioning of QGExp and QGComp operations with these order quasigroups is explained separately in detail.

3.1 Quasigroup Expansion (QGExp) Operation

Let each byte of data be divided into two 4-bit integers. That is, a character (one byte data) x is represented as $x = x_1x_0$, where x_0 and x_1 are 4-bit integers (hexadecimal

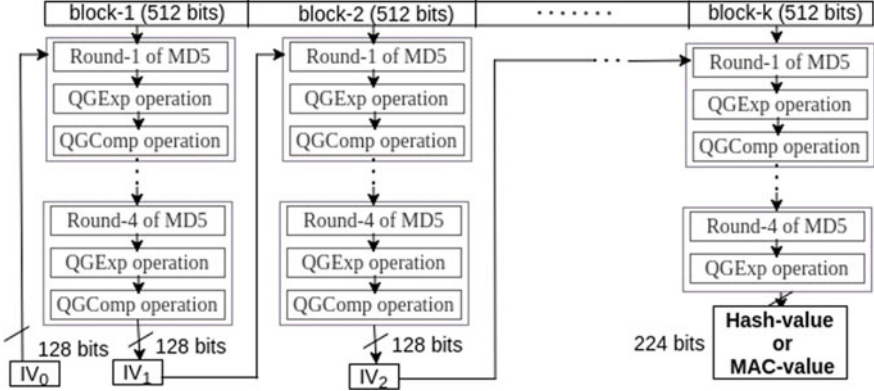


Fig. 1 Workflow of QGMDS and QGMAC

digits or nibble values). The QGExp operation takes two bytes of data and produces a sequence of three bytes of data. For the quasigroup of order 256, it is defined as

$$x_1x_0 \otimes_1 y_1y_0 = (x_1x_0, y_1y_0, z_1z_0), \quad (5)$$

where $z_1z_0 = x_1x_0 *_1 y_1y_0$, and \otimes_1 and $*_1$ are the QGExp operation and the quasigroup operation for the order 256, respectively. Note that z_1z_0 is the resultant element which is determined by looking up the element having the row index of x_1x_0 and the column index of y_1y_0 in the table representation of the quasigroup of order 256. And, for the quasigroup of order 16, it is defined as

$$x_1x_0 \otimes_2 y_1y_0 = (x_1x_0, y_1y_0, z_1||z_0), \quad (6)$$

where $z_1 = x_1 *_2 y_1$, $z_0 = x_0 *_2 y_0$, and \otimes_2 and $*_2$ are the QGExp operation and the quasigroup operation for the order 16, respectively, and $||$ is the concatenation operation that concatenates two 4 bits and makes it as one block of 8 bits. Note that z_1 is determined by looking up the element having the row index of x_1 and the column index of y_1 in the table representation of the quasigroup of order 16. Similarly, z_0 is determined by looking up the element having the row index of x_0 and the column index of y_0 in the table representation of the quasigroup of order 16.

An application of the QGExp operation to a pair of sequences of elements is as follows:

Let $A = (a_1^1 a_0^1, a_1^2 a_0^2, \dots, a_1^t a_0^t)$ and $B = (b_1^1 b_0^1, b_1^2 b_0^2, \dots, b_1^t b_0^t)$, where $a_1^i a_0^i$ and $b_1^j b_0^j$ are byte values whereas a_0^i , a_1^i , b_0^j , and b_1^j are nibble (4 bits) values, for $1 \leq i, j \leq t$, then

$$(A \otimes_1 B) \text{ or } (A \otimes_2 B) = ((a_1^1 a_0^1, b_1^1 b_0^1, r_1^1 r_0^1), (a_1^2 a_0^2, b_1^2 b_0^2, r_1^2 r_0^2), \dots, (a_1^t a_0^t, b_1^t b_0^t, r_1^t r_0^t))$$

where $r_1^j r_0^j = a_1^j a_0^j * b_1^j b_0^j$, $*$ is the quasigroup operation of order 256 with respect to the QGExp operation \otimes_1 or $r_1^j r_0^j = (a_1^j * b_1^j) || (a_0^j * b_0^j)$, $*$ is the quasigroup operation of order 16 with respect to the QGExp operation \otimes_2 and $||$ is the concatenation operation.

Similarly if $A = ((a_1^{11} a_0^{11}, a_1^{12} a_0^{12}, \dots, a_1^{1k} a_0^{1k}), (a_1^{21} a_0^{21}, a_1^{22} a_0^{22}, \dots, a_1^{2k} a_0^{2k}), \dots, (a_1^{t1} a_0^{t1}, a_1^{t2} a_0^{t2}, \dots, a_1^{tk} a_0^{tk}))$ and $B = (b_1^1 b_0^1, b_1^2 b_0^2, \dots, b_1^t b_0^t)$, where $a_1^{ij} a_0^{ij}$ is a byte value, a_0^{ij} and a_1^{ij} are nibble (4 bits) values for $1 \leq i \leq t$, $1 \leq j \leq k$, $b_1^l b_0^l$ is a byte value, b_0^l and b_1^l are nibble (4 bits) values for $1 \leq l \leq t$, then

$$\begin{aligned} (A \otimes_1 B) \text{ or } (A \otimes_2 B) &= ((a_1^{11} a_0^{11}, a_1^{12} a_0^{12}, \dots, a_1^{1k} a_0^{1k}, b_1^1 b_0^1, r_1^1 r_0^1), \\ &\quad (a_1^{21} a_0^{21}, a_1^{22} a_0^{22}, \dots, a_1^{2k} a_0^{2k}, b_1^2 b_0^2, r_1^2 r_0^2), \\ &\quad \dots, \\ &\quad (a_1^{t1} a_0^{t1}, a_1^{t2} a_0^{t2}, \dots, a_1^{tk} a_0^{tk}, b_1^t b_0^t, r_1^t r_0^t)) \end{aligned}$$

where $r_1^j r_0^j = a_1^{jk} a_0^{jk} * b_1^j b_0^j$, $*$ is the quasigroup operation of order 256 with respect to the QGExp operation \otimes_1 or $r_1^j r_0^j = (a_1^{jk} * b_1^j) || (a_0^{jk} * b_0^j)$, $*$ is the quasigroup operation of order 16 with respect to the QGExp operation \otimes_2 and $||$ is the concatenation operation.

3.2 Quasigroup Compression (QGComp) Operation

The QGComp operation compresses the partial hash-value (or MAC-value) of 224 bits into 128 bits. The resulting 128 bits are then fed into the next round of MD5 algorithm. The application of QGComp operation can be explained as follows: First it divides the 224 bits (28 byte) into 4 sub-blocks of 7 bytes each. It then operates on each of the 4 sub-blocks as follow: Let $A = (a_1^1 a_0^1, a_1^2 a_0^2, a_1^3 a_0^3, a_1^4 a_0^4, a_1^5 a_0^5, a_1^6 a_0^6, a_1^7 a_0^7)$ be a block of 7 byte, where $a_0^i a_1^i$ is a byte value, for $1 \leq i \leq 7$. Then, $\text{QGComp}(A) = (b_1^1 b_0^1, b_1^2 b_0^2, b_1^3 b_0^3, b_1^4 b_0^4)$, where $b_1^i b_0^i = a_1^i a_0^i * a_1^{8-i} a_0^{8-i}$, $*$ is the quasigroup operation of order 256 or $b_1^i b_0^i = (a_1^i * a_1^{8-i}) || (a_0^i * a_0^{8-i})$, $*$ is the quasigroup operation of order 16 for $1 \leq i \leq 3$ and $b_1^4 b_0^4 = a_1^4 a_0^4$.

4 Implementation and Software Performance

The proposed schemes have been implemented in C++ on a system that has the following configuration: Intel(R) Core(TM) i5-2400 CPU @3.40 GHz processor with 4 GB RAM and 64-bit Linux operating system. The source code of QGMD5, MD5, SHA-224, QGMAC, HMAC-MD5, and HMAC-SHA-224 is run 10^3 times for the message $M = \text{"The brown fox jumps over a lazy dog,"}$ and it calculated the average execution time in microseconds (μs). The C++ standard `<chrono>` library is used to

Table 3 Comparison of the average execution time for the message M in microseconds

Parameters	Hash functions			Message authentication codes		
	MD5	SHA-224	QGMD5	HMAC-MD5	HMAC-SHA-224	QGMAC
Avg. Exe. time (μs)	7.94	10.27	9.84	10.12	15.71	9.84

measure the execution time [6]. The performance of QGMD5 is compared with that of both MD5 and SHA-224 and the performance of QGMAC with that of both HMAC-MD5 and HMAC-SHA-224. The results of this analysis are presented in Table 3. Note that the average execution time of the proposed QGMD5 is $1.9 \mu s$ more than that of MD5 but not more than SHA-224. Also, note that the average execution time of the proposed QGMAC is always less than that of both HMAC-MD5 and HMAC-SHA-224. This is because the underlying structure of both QGMD5 and QGMAC is the same.

5 Security Analysis

5.1 Analysis of QGMD5

The proposed hash function was analyzed against the dictionary attack by subjecting its output to the online tools such as CrackStation [4] and HashCracker [7]. These tools are basically design to crack the hash-values of MD4, MD5, etc. They employ massive pre-computed lookup tables to crack password hashes. The proposed hash function is also analyzed and found to be resistant to various other attacks, including the brute-force attack. The strength of a hash function against the brute-force attack depends on the length of the hash-value produced by the hash function. The QGMD5 produces 224 bits hash-value instead of 128 bits, as in the case of MD5. Given an n bits hash-value brute-force attack to compute the pre-images (both first pre-image and second pre-image) requires 2^n effort, and to find a collision, it requires $2^{n/2}$ effort, where n is the size of the hash-value. Since the size of the hash-value of QGMD5 is 224 bits as against 128 bits of MD5, QGMD5 can be seen to be more secure than the MD5.

5.2 Collision Resistance

Collision resistance is an important property to test the security of a hash function because the space of messages and that of the hash values are related by a many-

to-one mapping. This means different messages may have the same hash-value. For this test, we randomly choose two messages M and M' , with hamming distance 1. We compute the hash values h and h' for each pair of messages M and M' and store in ASCII format (ASCII representation is a sequence of bytes in which each byte value lies from 0 to 255), then perform the following experiment [20]: Compare h and h' byte by byte and count the number of hits. That is, count the number of bytes that have the same value at the same position. In other words, compute

$$v = \sum_{p=i}^s f(d(x_p), d(x'_p)), \text{ where } f(x, y) = \begin{cases} 1, & x = y \\ 0, & x \neq y. \end{cases} \quad (7)$$

The function $d(\cdot)$ converts the entries to their equivalent decimal values and s denotes a number of bytes in a hash-value. Smaller v characterizes the stronger hash function against collision resistance.

Theoretically, for N independent experiments, the expected number of times v hits for an s bytes hash-value is calculated as follows:

$$W_N(v) = N \times Prob\{v\} = N \times \frac{s!}{v!(s-v)!} \left(\frac{1}{256}\right)^v \left(1 - \frac{1}{256}\right)^{s-v}, \quad (8)$$

where $v = 0, 1, 2, \dots, s$. A collision will never happen if $v = 0$, and a collision will happen if $v = s$. For $N = 2048$, we computed, using equation (8), the expected values of $W_N(v)$ for $s = 16$ and $s = 28$ byte hash-values, compared these results with those of the experimental values of MD5, SHA-224, and QGMD5, and tabulated these findings in Table 4. From the entries in Table 4, we observe that the experimental results of QGMD5 not only coincide very well with the theoretical ones but also it has the better collision resistance than that of both MD5 and SHA-224.

Table 4 Results of expected and experimental

v	Expected value of $W_N(v)$		Experimental value of $W_N(v)$		
	$s = 16$	$s = 28$	MD5 ($s = 16$)	SHA-224 ($s = 28$)	QGMD5 ($s = 28$), Pro.
0	1923.69	1835.42	1912	1828	1841
1	120.70	201.54	130	212	199
2	3.55	10.67	6	8	8
$v \geq 3$	0	0	0	0	0

5.3 Avalanche Effect

One of the desirable properties of a hash function is that it should exhibit a good avalanche effect. That is, for a slight change in the input difference, there should be a significant difference in the output of the hash function. The proposed hash function is tested for this property, and the resulting values are compared with those of MD5 and SHA-224. Details of the test are as follows: The message $M =$ “The brown fox jumps over a lazy dog” of 280 bits is randomly chosen, and 280 messages $(M_0, M_1, \dots, M_{279})$ are generated by changing the i^{th} bit in M and $0 \leq i \leq 279$.

Let $h = H(M)$ be the hash-value of the original message M and $h_i = H(M_i)$ be the hash-values of the messages M_i for $0 \leq i \leq 279$. Since the size of hash-value of MD5 is 128 bits and it differs from that of SHA-224 and QGMD5, the hamming distance h_i from h is measured in percentage using the following formula:

$$HDP_i = \frac{D(h, h_i)}{NB(h)} \times 100\% \quad (9)$$

where HDP_i denotes the hamming distance of h_i from h in percentage for $0 \leq i \leq 279$, $D(h, h_i)$ denotes the hamming distance between h and h_i , and $NB(h)$ denotes the total number of binary digits in hash-value h . Table 5 shows the number of times the hamming distances (HDP_i) of the hash-values h_0, h_1, \dots, h_{279} from h lie in the specified range for the hash functions MD5, SHA-224, and QGMD5. Also given in the table is the average (mean) of these values. From these values, we can conclude that the avalanche effect of QGMD5 is better than that of both MD5 and SHA-224.

5.4 Analysis of QGMAC

The security of the proposed message authentication code QGMAC depends on the hash function QGMD5 as well as on the quasigroup of order 256 that is used. This is because the quasigroup used in QGMAC acts as a secret key. Since the number of quasigroups of order 256 is lower bounded by 0.304×10^{101724} , it follows that the

Table 5 Hamming distances for MD5, SHA-224, and QGMD5

Range of HDP_i	Number of hash pairs of MD5	Number of hash pairs of SHA-224	Number of hash pairs of QGMD5 (proposed)
35–44.99	41	19	16
45–54.99	206	238	246
55–64.99	33	23	18
	Average hamming distance		
Mean:	49.76%	49.97%	50.02%

probability of identifying the chosen quasigroup is close to zero. Hence, QGMAC is resistant to brute-force attack. Also, QGMAC is resistant to forgery attack. In forgery attack, an attacker chooses a fixed n number of different messages (M_1, M_2, \dots, M_n) and their corresponding MAC-values (authentication tags) (h_1, h_2, \dots, h_n) and tries to solve the following equations for the key k :

$$h_i = H_k(M_i), \text{ for } 1 \leq i \leq n, \quad (10)$$

where, in our case, H is the QGMD5 and k is the quasigroup employed. This is because if the attacker can get the key, then the attacker can forge an authentication tag for any chosen message. But the above system of equations has as many solutions as there are quasigroups of order 256. Hence, determining the quasigroup makes it practically impossible. Therefore, the QGMAC is also resistant to forgery attack.

6 Conclusions

This paper has proposed an efficient method named QGMAC to compute the message authentication code of a message. This method is designed based on the concept called a quasigroup. This QGMAC uses the new hash function, named QGMD5, which is also proposed in this paper. The QGMD5 can be viewed as the extended version of MD5, and it uses the MD5 along with 16 optimal S-boxes of 4×4 bits that form an optimal quasigroup. Because of this, the relationship between the original message and the corresponding hash-value is not transparent. We have analyzed the QGMD5 by comparing it with both the MD5 and the SHA-244, including brute-force attack, collision resistance, and the avalanche effect. We observed that the QGMD5 is more secure than that of both MD5 and SHA-224. Also, the proposed QGMAC is analyzed against brute-force attack and forgery attack. We found that QGMAC is resistant to these attacks.

References

1. Denes J, Keedwell AD (1991) Latin squares: new developments in the theory and applications, vol. 46. Elsevier
2. Farhan D, Ali M (2015) Enhancement MD5 depend on multi techniques. Int J Softw Eng
3. Gupta DR (2020) A Review paper on concepts of cryptography and cryptographic hash function. Eur J Mol Clin Med 7(7):3397–408 Dec 24
4. <https://crackstation.net/crackstation-wordlist-password-cracking-dictionary.htm>
5. https://en.wikipedia.org/wiki/Dictionary_attack
6. <https://en.cppreference.com/w/cpp/chrono>
7. <https://www.onlinehashcrack.com>
8. Ilaiyaraja M, BalaMurugan P, Jayamala R (2014) Securing cloud data using cryptography with alert system. Int J Eng Res 3(3)

9. Jacobson MT, Matthews P (1996) Generating uniformly distributed random Latin squares. *J Combinator Des* 4(6):405–437
10. Leander G, Poschmann A (2007) On the classification of 4 bit S-Boxes. In: *Proceedings of the 1st international workshop on arithmetic of finite fields*. Springer, Berlin, pp 159–176
11. Maliberan EV, Sison AM, Medina RP (2018) A new approach in expanding the hash size of MD5. *Int J Commun Netw Inf Secur* 10(2):374–379
12. Meyer KA (2006) A new message authentication code based on the non-associativity of quasigroups
13. Mihajloska H, Gligoroski D (2012) Construction of optimal 4-bit S-boxes by quasigroups of order 4. In: *The sixth international conference on emerging security information, systems and technologies, SECURWARE*
14. Noura HN, Melki R, Chehab A, Fernandez Hernandez J (2020) Efficient and secure message authentication algorithm at the physical layer. *Wireless Netw* 9:1–5 Jun
15. Paar C, Pelzl J (2009) *Understanding cryptography: a textbook for students and practitioners*. Springer Science & Business Media
16. Rivest R (1992) The MD5 message-digest algorithm. RFC:1321
17. Selvi D, Velammal TG (2014) Modified method of generating randomized Latin squares. *IOSR J Comput Engi (IOSR-JCE)* 16:76–80
18. Stevens M (2007) Master's Thesis, On collisions for MD5
19. Theoharoulis K, Papaefstathiou I (2010) Implementing rainbow tables in high end FPGAs for superfast password cracking. In: *International conference on field programmable logic and applications*
20. Zhang J, Wang X, Zhang W (2007) Chaotic keyed hash function based on feedforward-feedback nonlinear digital filter. *Phys Lett A* 362(5–6):439–448

Leveraging Transfer Learning for Effective Recognition of Emotions from Images: A Review



Devangi Purkayastha and D. Malathi 

Abstract Emotions constitute an integral part of interpersonal communication and comprehending human behavior. Reliable analysis and interpretation of facial expressions are essential to gain a deeper insight into human behavior. Even though facial emotion recognition (FER) is extensively studied to improve human–computer interaction, it is yet elusive to human interpretation. Albeit humans have the innate capability to identify emotions through facial expressions, it is a challenging task to be accomplished by computer systems due to intra-class variations. While most of the recent works have performed well on datasets with images captured under controlled conditions, they fail to perform well on datasets that consist of variations in image lighting, shadows, facial orientation, noise, and partial faces. For all the tremendous performances of the existing works, there appears to be significant room for researchers. This paper emphasizes automatic FER on a single image for real-time emotion recognition using transfer learning. Since natural images suffer from problems of resolution, pose, and noise, this study proposes a deep learning approach based on transfer learning from a pre-trained VGG-16 network to significantly reduce training time and effort while achieving commendable improvement over previously proposed techniques and models on the FER-2013 dataset. The main contribution of this paper is to study and demonstrate the efficacy of multiple state-of-the-art models using transfer learning to conclude which is better to classify an input image as having one of the seven basic emotions: happy, sad, surprise, angry, disgust, fear, and neutral. The analysis shows that the VGG-16 model outperforms ResNet-50, DenseNet-121, EfficientNet-B2, and others while attaining a training accuracy of about 85% and validation accuracy as high as 67% in just 15 epochs with significantly lower training time.

D. Purkayastha (✉) · D. Malathi
Department of Computer Science and Engineering, SRM Institute of Science and Technology,
Kattankulathur 603203, India
e-mail: dp6014@srmist.edu.in

D. Malathi
e-mail: malathid@srmist.edu.in

Keywords Transfer learning · Emotion recognition · Convolutional neural networks · VGG-16 · ResNet-50 · DenseNet-121

1 Introduction

Achieving seamless and efficient interaction between next-generation computers and human is an envisaged ambition of artificial intelligence. The area of facial emotion recognition has been actively researched in the past few decades. Human emotions are expressed in a multitude of forms that are seldom perceptible to the naked eye. Emotion recognition can be performed using various features, including but not limited to face [1, 2], speech [3], EEG [4], and text [5]. Studies identified that around 60% to 80% of human communication is coming from nonverbal cues [6]. These signals include facial expressions, voice tone and pitch, eye contact, gestures, and physical distance. The facial expression is the most important input for analysis. Recognizing the emotion from the face image is the aim of facial expression recognition (FER). A major hurdle encountered is that the feature extraction process may be disturbed by the variance of the location of an object, noise, and lighting conditions of the image.

Using deep learning, particularly convolutional neural networks (CNNs), the facial expression recognition system can be developed with the features extracted and learned. In the course of recent years, several end-to-end frameworks have been proposed for FER using deep learning models as well as classical computer vision techniques. In FER, a majority of the indicators are selected from various parts of the face, viz. the eyes and mouth, whereas other parts, such as hair and ears, have little influence in the detection as stated in [7], where an attentional convolutional neural network has been proposed to study the most important parts of the face to perform FER tasks. Studies have shown that humans can classify seven emotions with an accuracy of approximately $65 \pm 5\%$ [8] in a face image. The complexity of this task can be observed when manually classifying the FER-2013 dataset images to following classes: {"*angry*", "*disgust*", "*fear*", "*happy*", "*sad*", "*surprise*", "*neutral*"}. Such tasks typically require the feature extractor to detect the features from an image, while the trained classifier produces the label(s) based on the features. Despite these challenges, modern AI systems are oriented to attend and solve tasks requiring robust and computationally inexpensive facial expression recognition. The facial expression recognition assists applications to achieve naturalistic interaction, improve responses, and better customization. In intelligent systems, learning and emotions are completely bound together; therefore, accurately identifying emotional states of learners could tremendously enhance the learning experience. Surveillance applications like driver monitoring systems and elderly monitoring systems could benefit by adapting to a person's cognitive status. Moreover, this could help monitor the treatment of patients undergoing medical treatment and understand their status better. In this work, a strategy dependent on transfer learning from VGG-16 is shown to outperform other architectures in FER tasks and can detect emotions

in the face images while achieving promising results with small datasets as well. This architecture is compared against various other architectures such as ResNet-50, DenseNet-201, and EfficientNet-B2.

The organization of the paper is as follows: In Sect. 2, previous works overview is given. In Sect. 3, the dataset, the methodology, and the experimentation procedure are described. Section 4 reports the observations, findings, and observations from analysis. Finally, Sect. 5 includes the concluding remarks while shedding some light on the prospects and research avenues of this work.

2 Contributions by Researchers on Human Facial Emotion Recognition

The human facial emotion recognition area has been well researched over the past two decades. This section gives a brief overview of the previous work done to perform FER tasks. A detailed survey of various approaches in every step can be referenced in [9]. Traditionally, algorithms for automatic facial expression recognition comprise three primary modules: image registration, feature extraction, and classification.

2.1 Feature Extraction Methods

Prior to the deep learning era, researchers depend upon hand engineered features such as scale invariant feature transformation (SIFT) [10], local binary patterns (LBP) [11], histogram of oriented gradients (HOG) [12], local phase quantization (LPQ) [13], histogram of optical flow [14], facial Landmarks [15, 16], Gabor wavelets [17], Haar features [18] as well as multiple PCA-based techniques [19] to successfully compute features from input images. Perhaps one of the most notable works performed in recognition of emotions is by Paul Ekman [20]. The sadness, happiness, anger, fear, disgust, and surprise were distinguished as the six principal emotions. Friesen et al. proposed facial action coding system (FACS) [21], depicting human facial expressions by their appearances on the face.

With the incredible achievement of deep learning [22], and particularly CNN for image classification and other vision problems, several groups proposed deep learning-based models for FER. To show a segment of the promising works, Lucey Patrick in [23] showed that convolutional neural networks can achieve high accuracy in recognition of emotions and used CNN with zero-bias on the Toronto Face Dataset (TFD) and the extended Cohn–Kanade dataset (CK+) to achieve state-of-the-art results. Shervin Minaee et al. in [7] proposed an attentional convolutional network, focusing on the feature-rich parts of the face while highlighting the most salient regions having the strongest impact on the classifier’s output. The entirety of the aforementioned works has achieved significant improvements over the normal

traditional works on recognition of emotions. To train a high-capacity classifier on smaller datasets, the technique of transfer learning has been widely employed where a network is initialized with the weights from a related task before fine-tuning them using a custom dataset. This approach has been proven to consistently achieve better results rather than training a network from scratch, and it is the method leveraged in this paper as well.

2.2 *Classification*

The feature extractor is usually followed by a classifier (support vector machine or ANN) which is trained on a set of videos or images used to detect the emotions. The classifier then assigns the emotion with the highest probability to the picture. For instance, [24] comprises a face detection module followed by a classification module that utilizes an ensemble of numerous deep CNNs. These methodologies appeared to turn out great on less difficult datasets, yet with the development of challenging datasets and in-the-wild images (having more intra-class variation), they started to exhibit their limitations. Since a large proportion of the features are hand-crafted for explicit applications, they are devoid of the required generalizability when fed with images having variations in pose, orientation, lighting, shadows, resolution, and noise.

2.3 *Transfer Learning*

In 2010, Pan et al. [25] introduced a method of “*learning unknown knowledge through existing knowledge*”. Transfer learning involves the concept of a domain and a task. It enables us to deal with scenarios of insufficient labeled or training data by leveraging the knowledge gained from pre-existing labeled data of some related task or domain to solve another task of a related domain.

3 **Methodology**

This section describes the dataset used, hardware specifications, data preprocessing applied, various architectures, and their performance comparison on FER after fine-tuning.

3.1 Dataset

In this work, the trial examination of the proposed approach was performed on the mainstream FER-2013 dataset from Kaggle. The Facial Expression Recognition 2013 database was first presented in the ICML 2013 Challenges in Representation Learning. The dataset contains 35,887 images of 48×48 resolution, a majority of which are taken in wild settings. The training set originally contained 28,709 images, while the validation and test set each comprised 3,589 images. In contrast with datasets such as CK+ [23], JAFFE [26], and FERF, this database has more intra-class variation in the images including partial faces, low-contrast, poor lighting, and face occlusion. This makes the dataset more challenging for FER tasks. The seven categories of emotions are labeled as: 0: Angry (4953 images), 1: Disgust (9547 images), 2: Fear (5121 images), 3: Happy (8989 images), 4: Neutral (6198 images), 5: Sad (6077 images), and 6: Surprise (4002 images).

3.2 Data Preprocessing

The FER-2013 dataset already consists of 48×48 grayscale images of faces that are almost centered with each face occupying about the same space in each image. The raw pixel data was normalized to lie between 0 and 1. To tackle the data imbalance problem, the technique of data augmentation was applied by applying random horizontal and vertical flipping to produce mirror images, zooming, rotations as well as height and width shifting.

3.3 Model Architectures

Convolutional Neural Network (CNN): It performs well on image-related tasks primarily because of two features:

1. Local receptive fields that learn correlations among neighborhood pixels.
2. Shared weights and biases that diminish the number of parameters to be learned, shifting invariance to the area under consideration.

VGG-Net: This network is characterized by small 3×3 convolutional layers with a stride of 1, arranged on top of each other in order of increasing depths and the volume reduction being done by max pooling layers. VGG consists of 2 fully connected layers, and each layer contains 4,096 nodes, followed by a softmax classifier. VGG-16 [27] contains 138 M parameters, close to 90% of which are in the last fully connected layer. More complex features can be learned through the 16 to 19 layers. Since the depth and amount of fully connected nodes, VGG-16 is over 533MB while VGG-19 is 574 MB, making its deployment an exhausting task, and it is shown in Fig. 1.

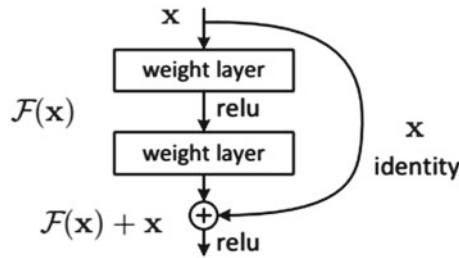


Fig. 1 Visualization of the VGG Architecture [28]

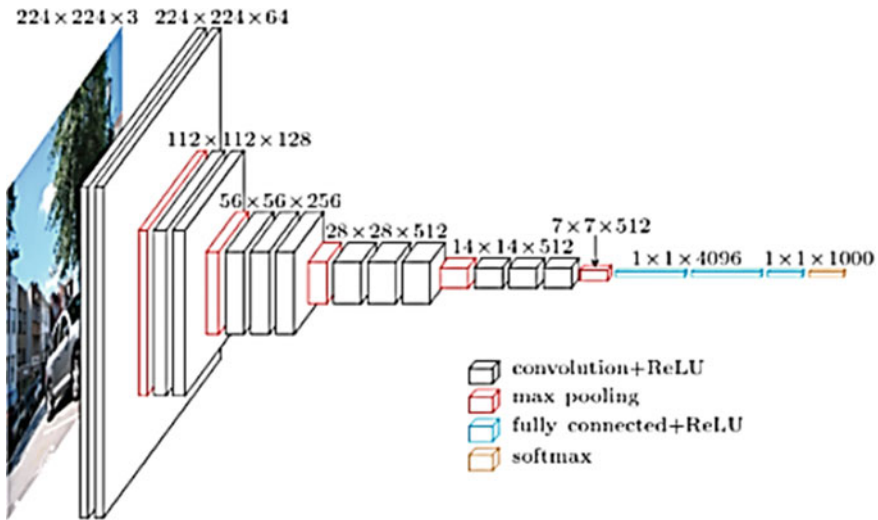


Fig. 2 Residual block as in [29]

ResNet: ResNet or residual network is a type of neural network based on small architecture modules or “network-in-network architectures”, which form the building blocks (in addition to convolutional and pooling layers), used to construct a macro-architecture. The ResNet [24] has 152 layers, eight times deeper than VGG-Net, having lower complexity, and by considering ImageNet dataset, it is managed to achieve 3.57% top-5 error. The authors of this paper introduced residual blocks given in Fig. 2; the identity function used in the block can be used as a shortcut during network optimization, facilitating the addition of multiple layers.

Inception and Exception: The recent architectures such as InceptionV3 [30, 31] as shown in Fig. 3 use scalar values to represent each feature map by considering the average of all elements in the feature map; this Global Average Pooling function minimizes the number of parameters in the last layers. This in turn compels the network to process input images to extract global features. The main feature of the

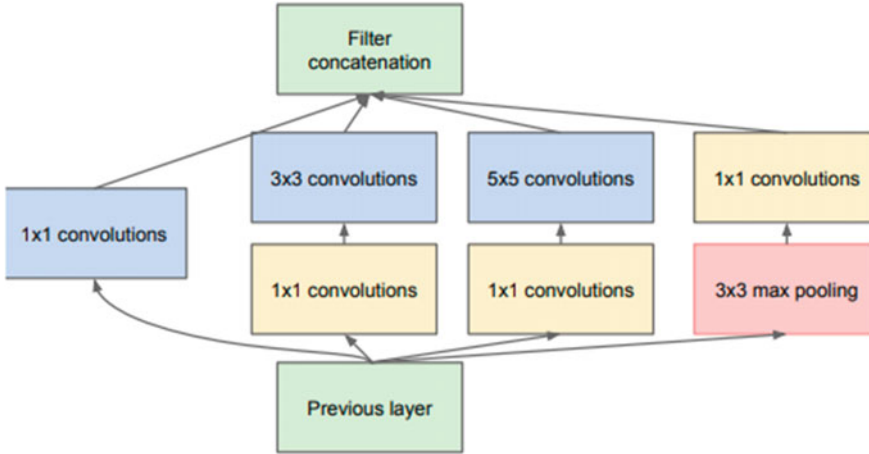


Fig. 3 Original Inception module used in Google_eNet [30]

Inception network [30] is the usage of multiple sizes of convolutional kernels such as (1×1) , (3×3) , and (5×5) to act as a “multi-level feature extractor”.

An expansion of the Inception architecture, the success of Xception [32] comes from the combination of depth-wise separable convolutions and residual modules. Depth-wise separable convolutions further diminish the number of parameters by isolating the two processes of feature extraction and combination within a convolutional layer. It has the smallest weight serialization: 91 MB (Fig. 4).

EfficientNet: The base network, EfficientNet-B0 as shown in Fig. 5 is based on the inverted bottleneck residual blocks of MobileNetV2, in addition to squeeze-and-excitation blocks. The critical component in EfficientNet [28] is the compound scaling method. EfficientNet-B7 achieves state-of-the-art 84.4% top-1 and 97.1% top-5 accuracy on ImageNet, while being $8.4\times$ smaller and $6.1\times$ faster on inference than the existing Convolutional Nets. Efficient Nets also transfer well and achieve state-of-the-art accuracy on CIFAR-100 (91.7%), Flower (98.8%), and three other transfer learning datasets, with an order of magnitude fewer parameters as stated in [25].

MobileNet-V2: MobileNetV2 [35], presented by Google, radically lessens the computational intricacy and model size of the network, making it the appropriate choice for devices with low computational power or mobile device. An inverted residual structure is the base of MobileNetV2. As shown in Fig. 5, it contains two different types of blocks: (i) The residual block with Stride 1. (ii) The block with Stride of 2 for downsizing. Both the blocks have three layers: 1×1 convolution with ReLU6, depth-wise convolution, and 1×1 convolution without nonlinearity (Fig. 6).

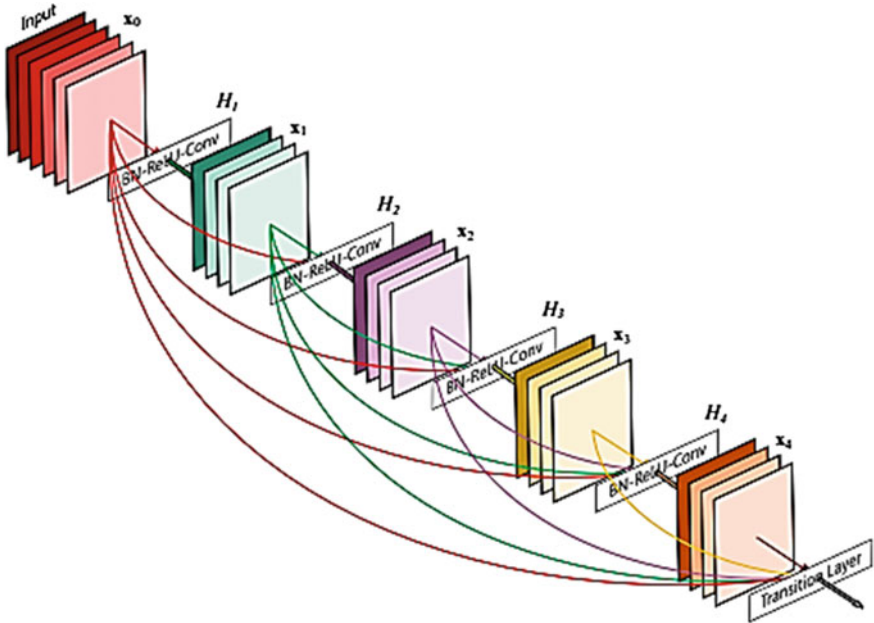


Fig. 4 Five-layer dense block with a growth rate of $k = 4$ [33]

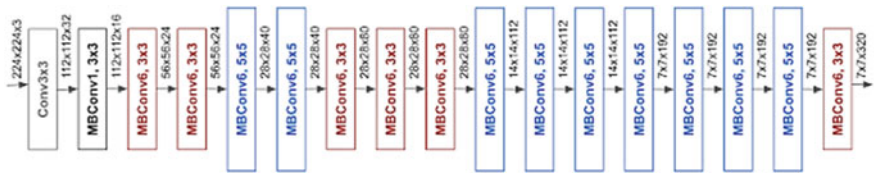
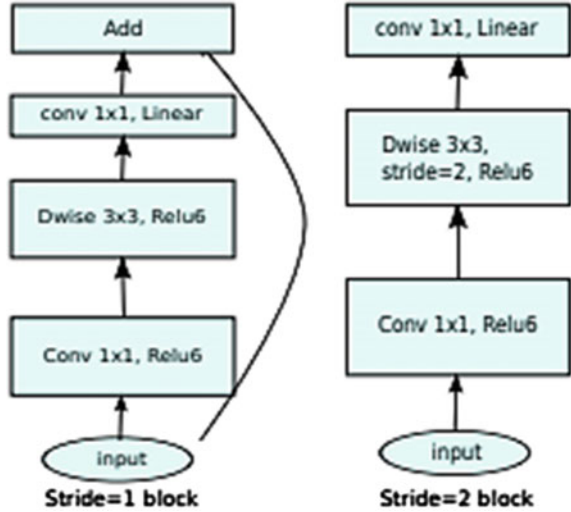


Fig. 5 Architecture for the baseline network EfficientNet-B0 [34]

3.4 Experimental Study

In this investigation, seven CNN architectures, viz. VGG-16, ResNet-50, DenseNet-121, EfficientNet-B2, MobileNet V2, Xception, and Inception-V3 have been analyzed in terms of their applicability and adequacy in facial emotion recognition and their accuracies have been compared. Execution and training of the aforementioned models have been done using the Keras high-level API and TensorFlow. GPU accelerated deep learning features were used to further speed up the model training process. For all the pre-training strategies employed, global average pooling has been applied at the last layer to diminish spatial dimensionality of information prior to passing it to the fully connected layers. Having experimented with different schemes for fine-tuning the base pre-trained CNN model on the FER-2013 dataset, it was tracked down that the models performed best when trained using the Adam optimizer with

Fig. 6 MobileNetV2 [35]



the accompanying hyperparameters: initial learning rate = 10^{-5} , epochs = 15, batch size = 256. Since the fine-tuning is performed on a relatively smaller dataset, the learning rate was subsequently changed to 10^{-4} to avoid radically altering the pre-trained weights and ran for 30 epochs with the same batch size of 256. The final output layer of the base model is taken out and supplanted by a Global Average Pooling layer. Loss function and categorical cross-entropy are given by Eqn. (1).

$$-\frac{1}{N} \sum_{i=1}^N \log(p(\text{model}(y_i) \in C_{y_i})) \quad (1)$$

where, $p_{\text{model}}(y_i \in C_{y_i})$ is the probability that y_i image belongs to category C_{y_i} .

4 Experimental Study and Comparison

It is observed that while ResNet-50 and Xception models perform decently well, EfficientNet-B2 has a highly erratic accuracy as well as loss curve. VGG-16 has significantly more parameters as compared to the other models, but has proven to outperform them on FER tasks. MobileNet struggles to achieve approximately 52% validation accuracy (Fig. 7; Table 1).

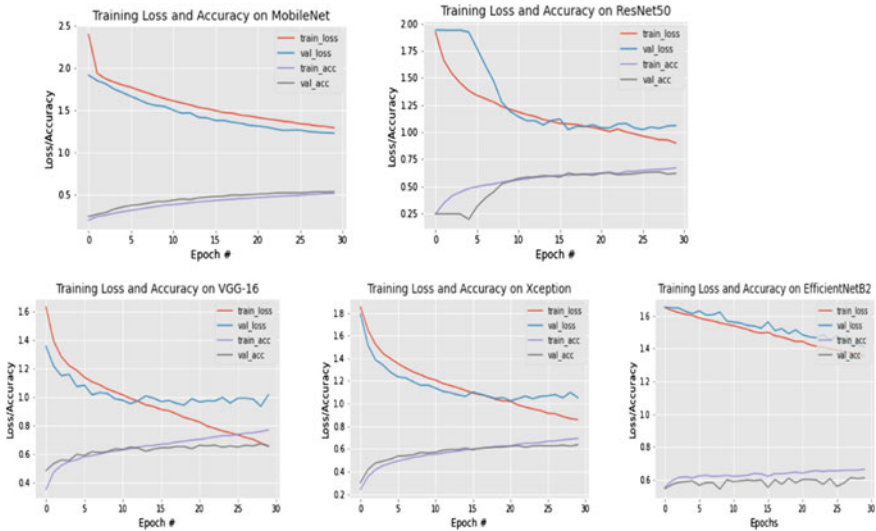


Fig. 7 Training loss and accuracy observed in various CNN architectures

Table 1 Comparison of the accuracy and validation accuracy of various CNN architectures, the time taken and number of parameters of each model for FER tasks

Model	Accuracy (%)	Val_Accuracy (%)	Time Taken (minutes)	Number of Parameters
VGG-16	87.44	67.02	24	138,357,544
ResNet-50	84.28	62.30	30	25,636,712
Xception	85.57	61.54	27	22,910,480
EfficientNet-B2	63.51	58.7	32	9,177,569
DenseNet-121	83.51	66.05	31	8,062,504
MobileNetV2	58.33	56.75	32	3,538,984
Inception-V3	68.74	60.3	32	23,851,784

5 Conclusion and Future Work

A transfer learning-based approach for building a real-time emotion recognition system has been presented while comparing the accuracies of various pre-trained CNN models and fine-tuning them on the FER-2013 dataset. This has been systematically developed to perform real-time inferences while significantly reducing training time and effort using modern architectures and advanced optimization methods. The image preprocessing and data augmentation techniques employed have been specified while providing a detailed account of the hyperparameters used for training. The performances of various state-of-the-art models like ResNet-50, VGG-16, EfficientNet-B2, DenseNet-121, Xception, and MobileNet-V2 have been compared and contrasted

for FER tasks, proving that VGG-16 shows the highest validation accuracy level of 67.02% after only 15 epochs while other models struggle to achieve a 62–63% validation accuracy even at the 30th epoch. It is encouraged to experiment with hybrid architectures, further fine-tune the hyperparameters and use visualization techniques to understand the high-level features learned by the model as well as discuss their interpretability. Furthermore, model biases may be explored to create more robust classifiers.

References

1. Mollahossein A, Chan D, Mahoor MH (2016) Going deeper in facial expression recognition using deep neural networks. In: Applications of computer vision (WACV), 2016 IEEE Winter Conference on IEEE
2. Ruvinga C, Malathi D, Dorathi Jayaseeli JD (2020) Human concentration level recognition based on vgg16 CNN architecture. *Int J Adv Sci Technol* 29(6s):1364–1373
3. Han K, Yu D, Tashev I (2014) Speech emotion recognition using deep neural network and extreme learning machine. In: Fifteenth annual conference of the International Speech Communication Association
4. Petrantonakis PC, Hadjileontiadias LJ (2010) Emotion recognition from EEG using higher order crossings. *IEEE Trans Inf Technol Biomed* 14(2):186–197
5. Chung-Hsien W, Ze-Jing C, Yu-Chung L (2006) Emotion recognition from text using semantic labels and separable mixture models. *ACM Trans Asian Lang Inf Process (TALIP)* 5(2):165–183
6. Mehrabian: nonverbal communication (1972) Aldine Transaction, New Brunswick
7. Minaee S, Abdolrashid A (2019) Deep-emotion: facial expression recognition using attentional convolutional network. *ArXiv*, abs/1902.01019
8. Goodfellow I et al (2013) Challenges in representation learning: a report on three machine learning contests
9. Sariyanidi E, Gunes H, Cavallaro A (2015) Automatic analysis of facial affect: a survey of registration, representation, and recognition. *IEEE Trans Pattern Anal Mach Intell* 37(5):1113–1133
10. Li Z, Imai JI, Kaneko M (2009) Facial-component-based bag of words and PHOG descriptor for facial expression recognition. In: Conference Proceedings—IEEE International Conference on Systems, Man and Cybernetics, pp. 1353–1358
11. Shan C, Gong S, McOwan PW (2009) Facial expression recognition based on local binary patterns: a comprehensive study. *Image Vis Comput* 27(5):803–816
12. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: IEEE computer society conference on computer vision and pattern recognition, vol 1, pp 886–893
13. Wang Z, Ying Z (2012) Facial expression recognition based on rotation invariant local phase quantization and sparse representation
14. Dalal N, Triggs B, Schmid C (2006) Human detection using oriented histograms of flow and appearance. In: European conference on computer vision, pp 428–441. Springer, Berlin
15. Cootes TF, Edwards GJ, Taylor CJ et al (2001) Active appearance models. *IEEE Trans Pattern Anal Mach Intell* 23(5):681–685
16. Cootes TF, Taylor CJ, Cooper DH, Graham J (1995) Active shape models-their training and application. *Comput Vis Image Understand* 61(1):38–59
17. Stewart BM, Littlewort G, Frank M, Lainscsek C, Fasel I, Movellan J (2005) Recognizing facial expression: machine learning and application to spontaneous behavior. In: IEEE Computer Society Conference on Computer vision and pattern recognition, vol 2, pp 568–573

18. Whitehill J, Omlin CW (2006) Haar features for faces au recognition. In: Automatic face and gesture recognition, FGR 2006. 7th International Conference, IEEE
19. Mohammadi M, Fatemizadeh E, Mahoor MH (2014) PCA based dictionary building for accurate facial expression recognition via sparse representation. *J Vis Commun Image Rep* 25(4):1082–1092
20. Paul E, Friesen Wallace V (1971) Constants across cultures in the face and emotion. *J Personal Soc Psychol* 17(2):124
21. Friesen E, Ekman P (1978) Facial action coding system: a technique for the measurement of facial movement. Palo Alto
22. Malathi D, Dorathi Jayaseeli JD, Gopika S, Senthil Kumar K (2017) Object recognition using the principles of Deep Learning Architecture. *ARPN J Eng Appl Sci* 12(12):3736–3739
23. Lucey P et al (2010) The extended Cohn-Kanade dataset (ck+): a complete dataset for action unit and emotion-specified expression. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE
24. Yu Z (2015) Image based static facial expression recognition with multiple deep network learning. In: ACM on international conference on multimodal interaction—ICMI, pp 435–442
25. Pan SJ, Yang Q (2010) A survey on transfer learning. *IEEE Trans Knowl Data Eng* 1345–1359 (2010)
26. Lyons MJ, Akamatsu S, Kamachi M, Gyoba J, Budynek J (1998) The Japanese female facial expression (JAFPE) database. In: Third international conference on automatic face and gesture recognition, pp 14–16
27. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556)
28. Tan, M, Le Q (2019) EfficientNet: rethinking model scaling for convolutional neural networks. PIn: roceedings of the 36th international conference on machine learning. In: Proceedings of machine learning research, vol 97, pp 6105–6114
29. 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 (CVPR), pp 770–778
30. Christian S, Vincent V (2015) Sergey Ioffe. In: Rethinking the inception architecture for computer vision. Jonathon Shlens
31. Szegedy C et al (2015) Going deeper with convolutions. In: 2015 IEEE conference on computer vision and pattern recognition (CVPR), Boston, MA, USA, pp 1–9
32. Chollet F (2016) Xception: deep learning with depth wise separable convolutions. CoRR. [abs/1610.02357](https://arxiv.org/abs/1610.02357)
33. Gao H, Zhuang L, van der Maaten L, Weinberger KQ (2016) Densely connected convolutional networks
34. <https://ai.googleblog.com/2019/05/efficientnet-improving-accuracy-and.html> . Last accessed on May 2021
35. Sandler M et al (2018) MobileNetV2: inverted residuals and linear bottlenecks. [http://arxiv.org/abs/1801.04381](https://arxiv.org/abs/1801.04381)

An Automated System for Facial Mask Detection and Face Recognition During COVID-19 Pandemic



Swati Shinde, Pragati Janjal, Gauri Pawar, Rutuja Rashinkar,
and Swapnil Rokade

Abstract The coronavirus (COVID-19) pandemic is an ongoing pandemic of coronavirus disease-2019. It is still spreading continuously across the globe, causing huge economic and social disruption. There are many measures that are suggested by the World Health Organization (WHO) to reduce the spread of this disease. In this paper, we are proposing a system in which people wear masks or not in public and recognize faces who do not wear masks. We detect the people who are monitored by using Webcam and those who are not wearing masks, and the corresponding authority is informed about the same by using convolutional neural network (CNN) with a mobile net and Haar cascade algorithm. The proposed model will help to reduce the spread of the virus and check the safety of surrounding people.

Keywords COVID-19 · Facial mask detection · Face recognition · Convolutional neural networks

1 Introduction

In 2019, the world faced a greater threat—coronavirus—the world is still facing it. Coronaviruses are a group of viruses that cause illness ranging from a simple cold to deadly infections like Severe Acute Respiratory Syndrome (SARS), Middle East Respiratory Syndrome (MERS), etc. [1]. In December of 2019, the first coronavirus case was detected. Since then, the number of coronavirus infected people has grown so rapidly that at present there are more than 60,721,235 cases out of which 1,426,843 people died due to the infection. These numbers are increasing on a daily basis. Most common and major symptoms of coronavirus are fever, tiredness, dry cough, pains and aches, headache, sore throat, loss of smell, or taste, etc.—as declared by the World Health Organization (WHO) [1].

S. Shinde (✉) · P. Janjal · G. Pawar · R. Rashinkar · S. Rokade
Computer Engineering Department, Pimpri Chinchwad College of Engineering, Nigdi, Pune
411044, India
e-mail: swaati.shinde@gmail.com

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
D. P. Agrawal et al. (eds.), *Cyber Security, Privacy and Networking*, Lecture Notes
in Networks and Systems 370, https://doi.org/10.1007/978-981-16-8664-1_3

25

Many precautionary measures are suggested to stop or at least reduce the spread of coronavirus. These measures include frequent cleaning of hands with soap, maintaining a safe distance from whoever is coughing or sneezing, wearing a mask in public, not touching your mouth, eyes, or nose, staying home if you are unwell, seeking medical attention before it is too late, etc.

It is observed that the spread of coronavirus can be reduced if people follow these precautionary measures. Wearing a mask in public and maintaining social distance being the most simple ones. But, it is found that people are too comfortable with their previous ways of living and show ignorance toward these simple measures which can save their lives. Thus, people's ignorance toward these safety measures is resulting in a speedy increase in the spread of coronavirus. To help reduce this rate of increase in the number of corona cases, a solution where a system detects if people are wearing masks or not and then identifying those who are not wearing and charging a fine can be of great help.

Facial mask detection is the detection of the presence of a mask on a person's face. It is very similar to object detection. Authorities like police cannot always keep watching people and then charge a fine if they are not wearing a mask. So, in such cases, some modern techniques like monitoring using Webcam along with the use of some deep learning algorithms to check if a person is wearing a mask or not can be convenient.

We are also performing face recognition. This is done to determine the identity of the unmasked person. It can help the authority a lot if he is able to keep a track of people who are constantly disobeying the rules that are made mandatory by the government.

In this paper, we are going to work on face mask detection and face recognition. In this, we have used convolutional neural network (CNN) for face mask detection purposes and Haar cascade algorithm for the face recognition module. For face mask detection, we have downloaded the dataset from Kaggle, and in face recognition, we created our own dataset.

2 Related Work

Since the outbreak of coronavirus (COVID-19), many researchers have studied the symptoms, preventive measures, etc., of coronavirus(COVID-19). Many have developed various models for various purposes which would help in controlling the spread of the virus. [1] presented a system for smart cities which was useful in reducing the spread of the virus. It was only determined whether people were masked or not. The model in [1] had some issues like it was not able to differentiate a masked face and a face covered with hands.

Another model [2] used a two-stage face mask detector. The first stage used a RetinaFace model for robust face detection. The second stage involved training three different lightweight face mask classifier models in the dataset. The NASNetMobile-based model was finally selected for classifying masked and unmasked faces.

MobileNet V2 architecture and computer vision is used to help maintain a safe environment and monitor public places to ensure the individual's safety [3]. Along with cameras for monitoring, a microcontroller has been used in [3].

Masked face recognition is done using the FaceNet pre-trained model in [4]. Three datasets are used in training the model in [4]. Recent work on this is coronavirus pandemic

(COVID-19): Emotional Toll Analysis on [5] Twitter. It is very important to perform the sentiment analysis of the tweet. By the use of this analysis, we are able to find the coronavirus (COVID-19) impact in people's life.

The online social media rumor identification is also there in that they used a parallel neural network for analysis [6].

This all is research or recent work done on the coronavirus (COVID-19).

3 Methodology

We proposed a system for controlling the spread of coronavirus. We are monitoring people in public places with the help of Webcam. With the help of these cameras, images are captured from public places. These captured images are then given as input to the proposed system. Then the system will detect whether a person is with a mask or without a mask, appearing in the image. If any person is found without a face mask, then his face is recognized and this information is sent to the respective authority office of that place. And if mass gathering is observed, then this information is also sent to respective authorities.

3.1 Image Preprocessing

To capture the real-time video footage of a person, the Webcam is used. To identify the person, images are extracted from video footage [7]. Before going to the next step, recognizing captured images by Webcam cameras required preprocessing. In this step, images are in RGB form, which contains a large amount of redundant information which is not required and RGB images stored 24 bits for each pixel. In the preprocessing step, RGB color images will transform into the grayscale color images because grayscale images will remove the unnecessary, redundant information, and it also stored the 8 bits for each pixel which is sufficient for the classification [8, 9]. Grayscale color images are reshaped uniformly; then, images are normalized in the range from 0 to 1. With the help of normalization, it captures the necessary features from the images and becomes faster.

3.2 Deep Learning Architecture

Deep learning is a very popular and powerful algorithm that learns different types of important features from the given samples. The nature of these features is nonlinear. To predict previously unseen samples, this architecture is used. To train this deep learning architecture, we collected the dataset from different sources. The architecture is dependent upon the convolutional neural network (CNN) [9, 10]. The deep learning architecture is explained below, in detail.

(i) Collection of the Dataset:

For the purpose of training and testing our deep learning model, we gathered some images, i.e., image datasets from open sources like kaggle, github, etc. The image dataset from kaggle contains 3833 images in total out of which 1915 images are with masks and the remaining, i.e., 1918 images, are without masks on the face. For training the first module which is the face mask detection module, we used 80% of the dataset and the rest was utilized for testing the module. For the second module, we took our own images for recognizing the person as shown in Resultset(B).

(ii) Development of the Architecture:

Our architecture is convolutional neural network (CNN) based. The reason behind this being that CNN automatically detects the important features from the images, without any human interference or supervision. Convolutional neural network (CNN) is also very useful in pattern recognition [11–13]. The network comprises mainly three types of layers: (a) input layer, (b) hidden layers, and (c) output layer. The input given is nothing but the image. In the second layer, which is the collection of hidden layers, in this, there are several convolutional layers which learn appropriate filters for extraction of features that are important. For the purpose of classification, the multiple dense neural networks use the features extracted from the hidden layers. In the architecture, three pairs of convolutional layers are followed by a max-pooling layer. This max-pooling layer helps in decreasing the spatial size of the representation and thus helps in reducing the number of parameters. This results in a simplified computation network. After this, a flatten layer is applied which converts the data into a one-dimensional array. This newly generated one-dimensional array is fed into the dense network. This dense network consists of three pairs of dense layers and dropout layers which learn parameters that are useful for classification. The dense layer consists of a series of neurons. These neurons learn the nonlinear features. The job of dropout layers is to prevent overfitting, and this is done by dropping some of the units.

Figure 1 a, b shows the block diagram of the model proposed in this paper.

For detecting a person with a mask and without a mask, accordingly, we trained our module with thousands of images. Basically, in this module, we are going to follow the convolutional neural network (CNN), but there is a small change in it as shown in Fig. 2. The basic idea to implement in this system is that we are going to

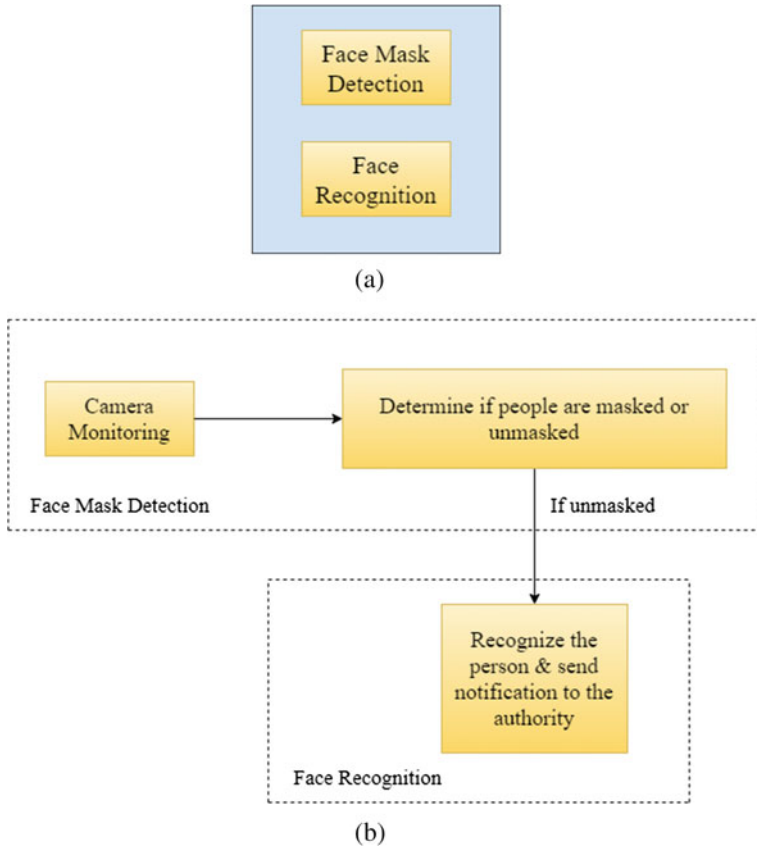


Fig. 1 Block diagram of the proposed model

neglect convolution that we usually do or use for image preprocessing and instead of that we introduce here MobileNet V2 because it is a fast and powerful module.

The trained module is applied in faceNet for detecting the faces because it contains the couple of files that we had for face detection.

The above image shows training images of face mask detection modules for detecting if that person is wearing a mask or not. The dataset has been downloaded from kaggle.

3.3 Face Recognition Module

For face recognition, first we have to detect if the person is wearing a face mask or not. If the person is not wearing a mask, then detect that person. In this module, we

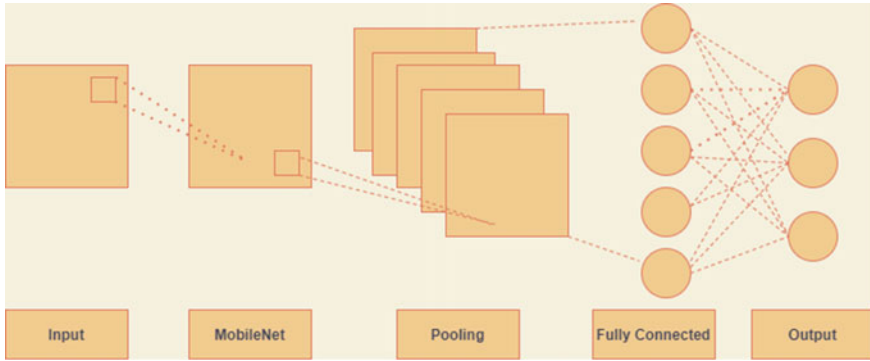


Fig. 2 MobileNet V2 in our system

are going to follow the herd cascade algorithm. Firstly, we have created a dataset of the people. After that we need to train the module.

The trained module is applied in a system of haar cascade algorithms to detect the face of a person.

Figure 4 is for training datasets, images of face detection. This dataset we have created for the recognition or detection of a person.

4 Algorithm Used in Proposed model

4.1 Convolutional Neural Network (CNN)

In the proposed model, we are using the convolutional neural network (CNN) architecture because convolutional neural network (CNN) is very useful in pattern recognition and also detects features from given images.

Layers in Convolutional Neural Network (CNN):

- (a) Input layer,
- (b) Hidden layers, and
- (c) Output layer.

Input layer:

The input layer in CNN should contain only images and reshape into a single column.

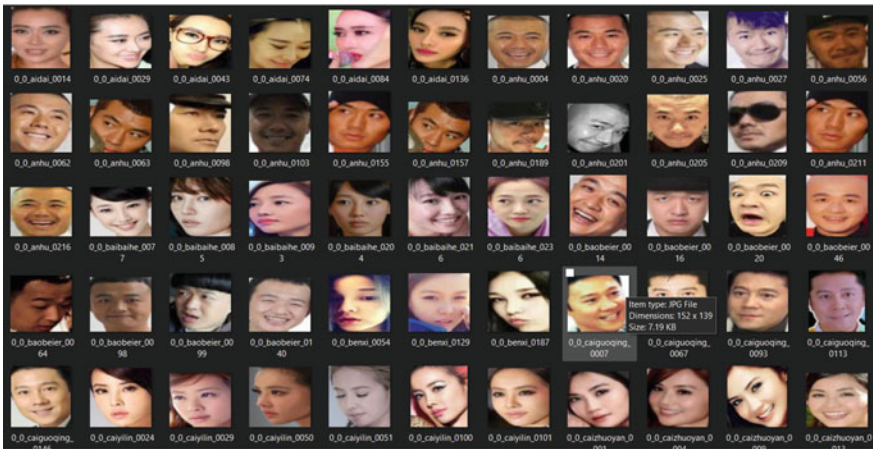
Hidden layer:

The hidden layer is nothing but a nonlinear transformation of the input which applies weights to the input, and along with all layers, we are using mobile net v2 because it is more powerful.

Output layer:



(a)



(b)

Fig. 3 a Dataset for mask recognition, b dataset for unmask detection

The output layer contains the label which is in the form of one-hot encoded. This max-pooling layer helps in decreasing the spatial size of the representation and thus helps in reducing the number of parameters.



Fig. 4 Training dataset image of face detection

4.2 Haar Cascade Algorithm

Haar cascade is an object detection algorithm. This algorithm is used to identify the faces in real-time video. This algorithm uses line detection or edge detection features. To detect the edge of the image her feature traverses the whole image. This algorithm uses line detection or edge detection features to detect the edge of the image haar feature traverses the whole image.

This haar features traverses from top left of the image to the bottom right of the image. This hour feature is good at detecting the edges and lines of the image. Because of that it is an effective feature of face detection.

Haar cascade is used in this face detection module because this algorithm is specially designed for detecting the object. It can be used for detecting the faces in the videos. So that this hair cascade algorithm is best suitable for this module [14].

5 Limitations and Future Works

The proposed system will identify the person without a mask. If a person is found without a mask, then information will be sent to respective authorities. Based on this information, the authority will find out the person and take necessary action. Only one limitation is there in a system; that is, we require the dataset of a person in our system to identify the person. That is why the system is useful for any organization, institute, school, and college.



Fig. 5 Identified masked and unmasked faces

6 RESULTS

6.1 Face Mask Detection Module

We have trained our module with thousands of masked and unmasked images so after testing on the video stream of the Webcam, it shows some results like shown in Fig. 5.

In Fig. 5a as the person is with a mask on his face, our model detects him as a masked image and gives a green color bounding box with ‘Mask’ tag. The same person is present without a mask in Fig. 5b so the model tagged him with a ‘NO Mask’ bounding box. This time the bounding box color turned red.

6.2 Face Recognition Module

The face mask detection module does the task for determining if people are masked or unmasked. But, as we have to notify authorities, we need to identify people who are not following the preventive measures.

This task of finding who the person was is done in the face recognition module. Figure 6a, b shows the same. In Fig. 6a, an unknown person is in front of the camera. Thus, he has been tagged as ‘Unknown’, whereas in Fig. 6b, the person is recognized as the name ‘Swapnil’ because we have trained our model with the images of Swapnil’s face and given the name tag as ‘Swapnil’.

In this face recognition module, first we need to create a database of the person for the identification. If the database is present, then only the system is able to identify the person. Otherwise, it will show the person is ‘Unknown.’

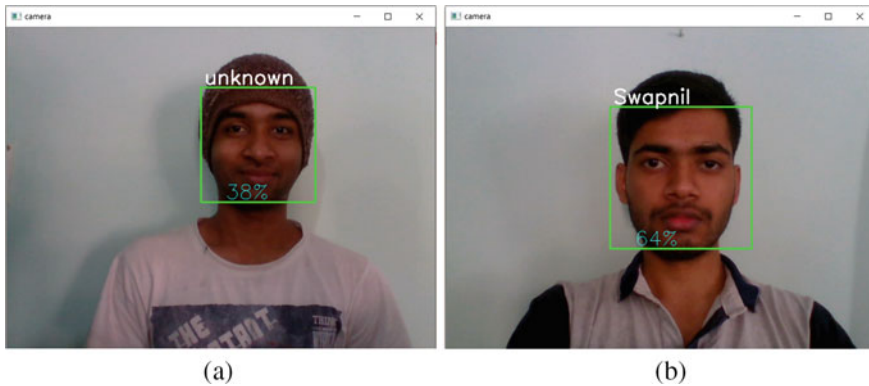


Fig. 6 Identified unmasked train image face

7 Conclusion

In this paper, we proposed an approach that uses MobileNet V2 and Haar cascade algorithm to face mask detection and face detection. In this COVID-19 situation, it is really important that we have to follow all the guidelines given by the government. For that purpose, we have developed this model to detect whether people are wearing masks or not. If people are not wearing them, then detect the person and show the name of that person. We have successfully developed a module that detects whether people are wearing masks or not. If not, then detect that person. This module can be used in organization, schools, colleges, etc.

References

1. Rahman M, Manik MMH, Islam M, Mahmud S, Kim J-H (2020) An automated system to limit COVID-19 using facial mask detection in smart city network. <https://doi.org/10.1109/IEMTRONICS51293.2020.9216386>
2. Chavda A, Dsouza J, Badgujar S, Damani A (2020) Multi-stage CNN architecture for face mask detection
3. Yadav S (2020) Deep learning based safe social distancing and face mask detection in public areas for COVID-19 safety guidelines adherence. *Int J Res Appl Sci Eng Technol* 8:1368–1375. <https://doi.org/10.22214/ijraset.2020.30560>
4. Ejaz M, Islam M (2019) Masked face recognition using convolutional neural network, pp 1–6. <https://doi.org/10.1109/STI47673.2019.9068044>
5. https://www.researchgate.net/publication/350982977_Coronavirus_Pandemic_COVID-19_Emotional_Toll_Analysis_on_Twitter
6. <https://www.igi-global.com/article/a-parallel-neural-network-approach-for-faster-rumor-identification-in-online-social-networks/240236>
7. Hire M, Shinde S (2018) Ant colony optimization based exudates segmentation in retinal fundus images and classification. In: 2018 Fourth international conference on computing communication control and automation (ICCUBEA). <https://doi.org/10.1109/ICCUBEA.2018.8697727>

8. Somvanshi M, Chavan P, Tambade S, Shinde SV (2016) A review of machine learning techniques using decision tree and support vector machine. In: 2016 International conference on computing communication control and automation (ICCUBEA). <https://doi.org/10.1109/ICCUBEA.2016.7860040>
9. Patil C, Shinde S (2019) Leaf detection by extracting leaf features with convolutional neural network, 18 May 2019. In: Proceedings of international conference on communication and information processing (ICCIIP) 2019. Available at SSRN: <https://ssrn.com/abstract=3419766> or <https://doi.org/10.2139/ssrn.3419766>
10. Deshmukh S, Shinde S (2016) Diagnosis of lung cancer using pruned fuzzy min-max neural network. In: 2016 International conference on automatic control and dynamic optimization techniques (ICACDOT), Pune, 2016, pp 398–402. <https://doi.org/10.1109/ICACDOT.2016.7877616>
11. Mane S, Shinde S (2018) A method for melanoma skin cancer detection using dermoscopy images. In: 2018 Fourth international conference on computing communication control and automation (ICCUBEA)
12. Ge S, Li J, Ye Q, Luo Z (2017) Detecting masked faces in the wild with LLE-CNNs. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR), Honolulu, HI, 2017, pp 426–434. <https://doi.org/10.1109/CVPR.2017.53>
13. Jiang M, Fan X (2020) RetinaMask: a face mask detector
14. Ejaz MS, Islam MR, Sifatullah M, Sarker A (2019) Implementation of principal component analysis on masked and unmasked face recognition. In: 2019 1st International conference on advances in science, engineering and robotics technology (ICASERT), Dhaka, Bangladesh, 2019, pp 1–5. <https://doi.org/10.1109/ICASERT.2019.8934543>

ROS Simulation-Based Autonomous Navigation Systems and Object Detection



Swati Shinde, Tanvi Mahajan, Suyash Khachane, Saurabh Kulkarni,
and Prasad Borle

Abstract Autonomous robots are becoming popular and are being used in many industries, due to their autonomy features. They are just like humans who have the ability to make decisions on their own without any human help. As the need for such robots is increasing, our paper aims to present an ROS autonomous navigation software system for autonomous robots, which is capable of creating 2D and 3D maps of the Simulation environment, localizing the robot in that environment and further performing path planning of the robot along with object detection using ROS. Moreover, various algorithms used for creating maps along with detailed internal working of the packages used for path planning are being discussed in this paper.

Keywords ROS · Turtlebot 2 · Autonomous navigation · Gmappinga · RTAB-map · Object detection

1 Introduction

Nowadays, robots are becoming more and more popular. Many industries have also started adopting robots for their quotidian works. Robots can be of various types like humanoid robots, teleoperated robots, autonomous robots. Each robot is used for different purposes. Autonomous robots, also called Autobots, perform tasks on their own without any human help. They just sense the environment with the help of sensors like cameras, lasers, infrared sensors, and then this sensed information is been processed which further helps them in navigating, avoiding obstacles or performing some specific tasks on their own. Thus, this paper mainly focuses on autonomous robots and aims to explain simulation of autonomous navigation on ROS robot Turtlebot 2. The autonomous navigation system has been integrated with ROS. This system is capable of creating 2D and 3D maps of the environment using SLAM algorithms and also performs object detection and navigation.

S. Shinde (✉) · T. Mahajan · S. Khachane · S. Kulkarni · P. Borle
Pimpri Chinchwad College of Engineering, Pune, Maharashtra 411035, India
e-mail: swaatti.shinde@gmail.com

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
D. P. Agrawal et al. (eds.), *Cyber Security, Privacy and Networking*, Lecture Notes
in Networks and Systems 370, https://doi.org/10.1007/978-981-16-8664-1_4

Firstly, the paper explains the robot and the environment used for the simulation, the software tools and platform used. Secondly, the ROS autonomous navigation system has been elaborated, followed by object detection in the environment. Along with this, the results are evaluated.

2 Related Work

In [1], the authors have presented autonomous navigation of robot using Gmapping algorithm with the help of ROS. The robot was simulated on a Gazebo environment where SLAM and navigation were performed. Results say that the map of environment was successfully created, and the robot was able to perform navigation without colliding [2].

In research paper [3], authors have built an autonomous navigation platform [3] where the robot is able to create maps of environment and further navigate. They have designed this for indoor applications. Gmapping algorithm was used for the mapping process, and they have also evaluated the results with respect to mapping, localization and navigation. Their study says that the robot gives good response time [3] and reasonable time [3] to navigate from one point to another.

3 Robot and Environment

Turtlebot 2 which is the second generation of Turtlebot robot [4] is been for testing the algorithms for mapping and object detection in a simulated environment. It is a low-cost mobile robot used for education and research [4]. It contains a kukubi base, kinect mounting hardware, asus xtion pro live, etc. (Fig. 1).

Fig. 1 Turtlebot 2 [2]





Fig. 2 Warehouse environment

Simulation environment is just a virtually created environment which can be used for testing of any application. The warehouse environment which consists of a conveyor belt, some wooden blocks and wooden storage area has been provided by RDS as shown in Fig. 2. This environment is being used for testing the robot and algorithms for mapping and navigation.

4 Software and Platforms

4.1 ROS

ROS is a robot operating system [5] is used as a middleware which provides libraries and tools to create robot applications. Running processes on ROS take place in the form of nodes, which may receive, post and multiplex sensor data, control messages passing between different nodes. ROS is an open-source platform [5] that supports various operating systems, but it is not a real operating system. It provides licensed packages, tools and libraries that are used to implement functionality of robot models, hardware drivers, planning, simulation tools and slam algorithms.

4.2 RDS

RDS is a platform provided by the construct. It helps programmers to program their robots, test the programs in real time [6] and simulate environments like warehouse environments, empty environments and golf environments. It also provides graphical tools like Rviz and rqt and robots like Turtlebot 2, Husky and R2C.

4.3 RVIZ

Rviz is a 3D visualization tool for the ROS framework. It helps us to view our robot and display robot's sensor data like laser scans, images captured by camera and Imu data. Maps created by the robot are visualized in Rviz. All the visualization processes, i.e. mapping, localization and navigation, are also visualized in Rviz.

5 ROS Autonomous Navigation

Robot navigation is about making a robot move autonomously in an environment. There are many techniques to make robots navigate. The most common is based on SLAM. Robot navigation [6] is elaborated in below steps:

1. Map Creation : Creating a map of the environment in which the robot will navigate.
2. Localization : It tells the current position of the robot in the map created.
3. Path Planning : To give a path for the robot to reach its goal position.

5.1 Map Creation

Robots must be familiar with the environment, and for this, the robot must be able to create a map of it. Maps are nothing but a representation of the environment where the robot will be navigating autonomously. SLAM helps achieve this task. SLAM is simultaneous localization and mapping. Within ROS, the most commonly used and widely accepted SLAM nodes are Gmapping & Hector SLAM. Some SLAM mapping algorithms are discussed below.

Gmapping. Gmapping is a ROS-based SLAM algorithm used for the creation of 2D maps of the environment. It takes inputs as the laser scanned data and odometry. ROS provides the `slam_gmapping` node [7] from the Gmapping package for mapping. This node subscribes to the laser scanned data, odometry, “/tf” topic and publishes “/map” topic. Once all parameters are tuned of `slam_gmapping` node, the map can be visualized in the rviz tool.

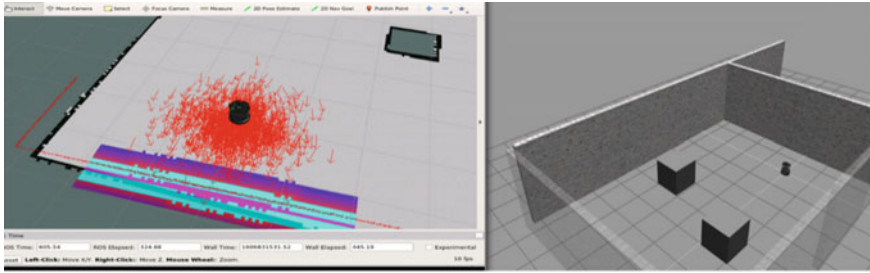


Fig. 3 Localization of robot

Hector Slam. Hector SLAM [8] takes advantage of the high update rate of laser scanners to provide an accurate estimate of the robot’s location and pose while mapping the environment [8]. Hector SLAM does not require odometry for creating a map of the environment. It is potential enough to build a map using only laser scanned data.

RTAB-Map. RTAB-map is Real-Time Appearance-Based Mapping [9], a 3D slam algorithm used for creating 3D maps of the environment. RTAB-map takes laser scanned data along with odometry information to frame pose. Loop closure detection [10] ensures that the robot has visited that place or not and adds a new constraint to the map’s graph. Graph optimizer is used to minimize error in the map. With generation 3D point clouds of environment RTAB-map 3D map.

5.2 Localization

An autonomous robot should know the environment and where it is in the environment. Localization means finding out the location or position of a robot in a given environment provided he has visited the environment at least once. ROS package AMCL helps localize a robot in the environment.

AMCL is an adaptive Monte Carlo localization approach, a probabilistic localization system for a robot moving in a 2D environment [11]. It takes input from the laser scan data, /tf topic, initial position of the robot, and most important is the 2D map. It publishes the robot’s estimated pose on the map. In the rviz tool, we can visualize this and we can localize the robot in the rviz tool using 2D pose estimation by just specifying the position and direction of the robot facing towards.

In Fig. 3, we can see that the red arrows are indicating that the robot is facing towards the wall. This is the localization of Turtlebot 2 using AMCL.

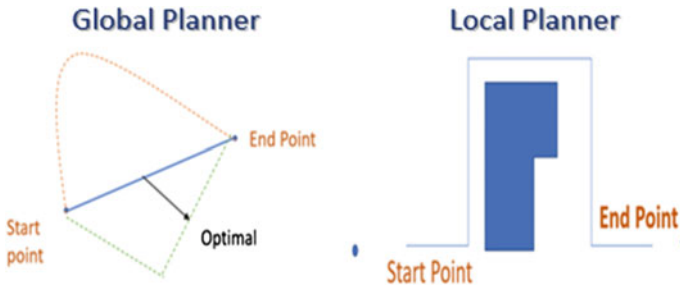


Fig. 4 Global planner and local planner

5.3 Path Planning

Path planning is nothing but it determines the sequence of steps the robot must take from current position to give goal or desired position. While planning the path, it takes into consideration the obstacles which are in the path. ROS has a package named `move_base` which helps to plan the path. This package contains the `move_base` node which acts as the path planner. `Move_base` node creates the costmaps based on the given map as input. Navigation is performed with the help `move_base` package which uses the concept of global planner and local planner.

Costmaps. Costmaps are the places which are safe for the robot. There are two costmaps—local costmaps and global costmaps. Local costmaps take the dynamic maps which take the sensor data reading into consideration which is used for getting the current motion of the robot, while the global costmaps are static maps which are static.

Global Planner. Global planner is used to calculate a path. It calculates a safe path for the robot based on the static map of the environment provided. Global planner uses the global costmap. When a goal is sent to the robot the `move_base` sends that goal to the global planner. Based on that goal, a path is calculated. For calculating paths, there are three global planners available : `global_planner`, `Navfn` [12], `carrot_planner`. `Navfn` is the most commonly used global planner. It internally uses the Dijkstra algorithm.

Local Planner. Local planner uses the local costmaps. When the path is calculated by a global planner, it is sent to the local planner. Local planner, based on the path given by global planner, sends commands to robots on `/cmd_vel` topic to move to the goal position. While approaching the goal position, it takes runtime sensor reading data into consideration, which means if any runtime obstacle comes into the environment which was not detected in the global costmap as while building the map it wasn't present, that is been detected in the local costmap. If any obstacle is detected by the local planner, then it recalculates the path for moving towards the goal. To achieve this, we need to tune parameters files of the `move_base` package.

Three basic parameter files are there—the global parameters file, local parameter file and common parameters file. All these are “.yaml” files (Fig. 4).

6 Object Detection

During navigation of the robot, obstacles in the robot’s path can be handled by avoiding it using object detection results. YOLO ROS: real-time object detection for ROS, provides `darknet_ros` [13] a ROS-based packet for object detection for robots. DarkNet is an open source, fast, accurate neural network framework used with YOLOv3 [14] for object detection as it provides higher speed due to GPU computations. YOLOv3 is the real-time object detection technique which uses trained image weights [15] for detection of new objects which is provided by this package. This will take the camera topic as input, and it will publish the image detected, where the detected object will be bound by boxes. Moreover, it will also give us the accuracy match percentage of the objects detected.

7 Results

7.1 Room Map Creation

The map creation was performed on the warehouse environment, Fig. 5 shows a 3D map created, where the actual objects, walls and things placed in the environment are seen in the map using the RTAB-map algorithm, and Fig. 6 shows the 2D map created of the environment with Gmapping SLAM algorithm. From results, we can see that accurate maps were been built. Hence, Gmapping and RTAB-map have performed well in the warehouse environment with Turtlebot 2.

7.2 Object Detection

We have performed object detection in the object placed environment as shown in Fig. 7. This environment is created in gazebo, for the purpose of object detection where various objects like sofa, stop sign and laptop are been placed. With the help `darknet_ros` package, the person, traffic lights, stop sign, sofa and chair are being detected by Turtlebot 2 as shown in Fig. 8. Sofa was been detected with an 100% accuracy match, traffic light and stop sign with an accuracy match of 98%. Therefore, `darknet_ros` package has performed very well with Turtlebot 2 and object placed environment.

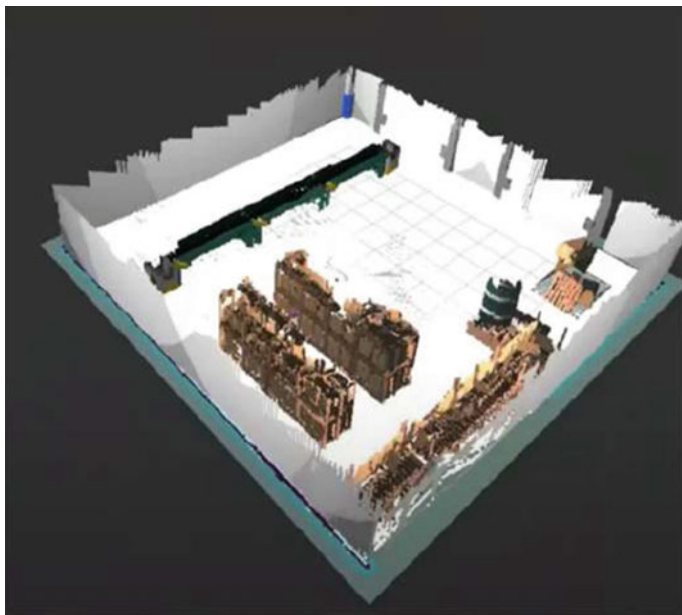


Fig. 5 RTAB-Map

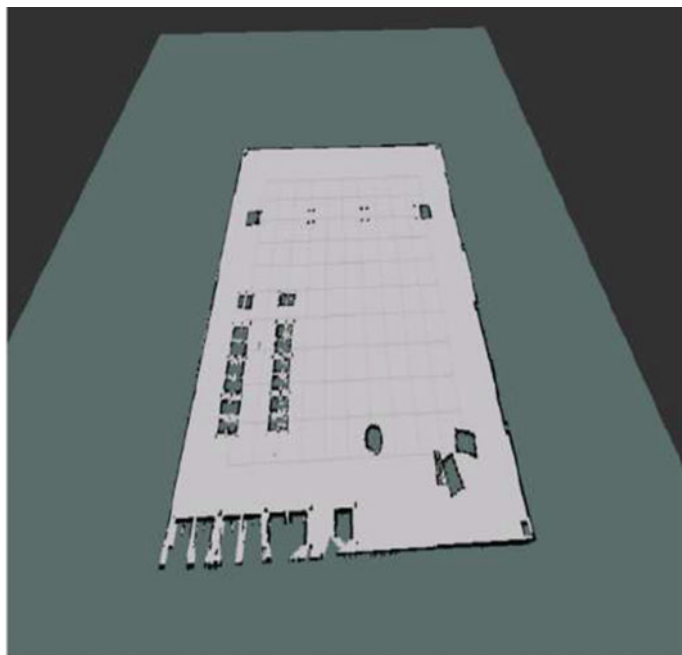


Fig. 6 Gmapping map



Fig. 7 Environment of object

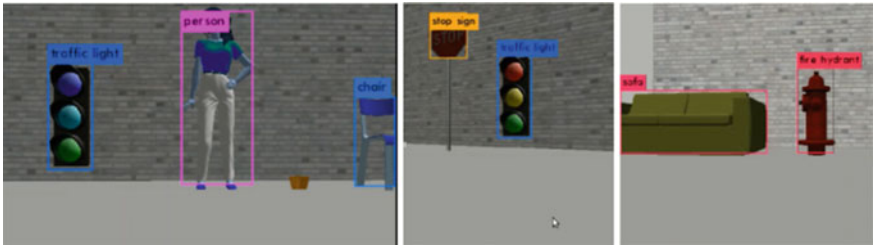


Fig. 8 Object detected in environment

7.3 Navigation

Turtlebot 2 has performed 2D navigation on the warehouse environment. The map created using the Gmapping algorithm as shown in, i.e. Fig. 6, was passed to the navigation system. With the help of AMCL package, the robot was able to localize itself in the environment. After giving a goal position to the robot, it moves towards the goal position with the help of global and local planner. The green line in Fig. 9 indicates the path of the robot to move to the desired goal. Robot was able to move to the desired goal position in good interval time.

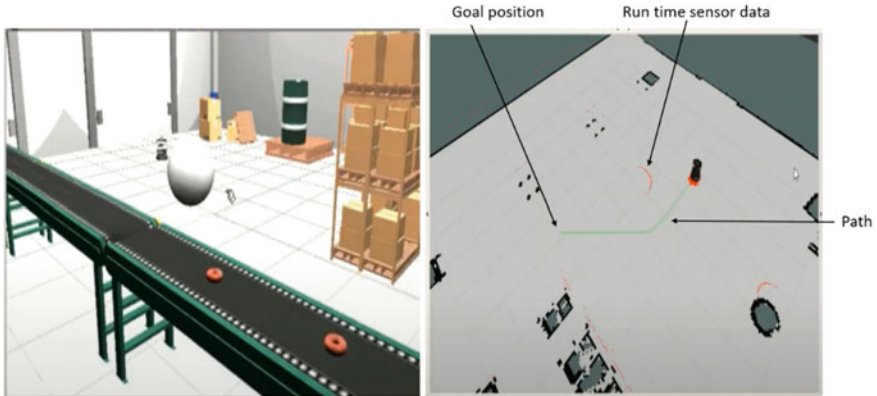


Fig. 9 Robot Navigation

8 Conclusion and Further Work

In this paper, we have presented the navigation software system for Turtlebot 2, where 2D maps using Gmapping and 3D maps using RTAB-map were built. Further, Turtlebot 2 was able to navigate in the known warehouse environment when a desired goal was given, with the help of ROS move_base package. Along with this, object detection was also performed on an object placed environment, where darknet_ros package was used. Turtlebot 2 was successfully able to detect the objects that were placed in that environment. Moreover, this study can be taken further where the warehouse environment and the object placed environment can be integrated where creating 3D maps along with object detection can take place at the same time.

References

1. Sumegh T, Mihir P, Pranjali, T Pratik K (2020) ROS based navigation using Turtlebot. ITM Web of Conferences, vol 32, p 01011
2. Thale SP, Prabhu MM, Thakur PV, Kadam P (2020) ROS based SLAM implementation for autonomous navigation using Turtlebot. In: ITM Web Conference, vol 32, p 01011
3. Megalingam RK, Chinta R, Sreekant S, Raj A (2019) ROS based autonomous indoor navigation simulation using SLAM algorithm
4. Turtlebot2, <https://www.turtlebot.com/turtlebot2/>
5. ROS, <http://wiki.ros.org/ROS/Introduction>
6. The ROS development studio by the construct. <https://www.theconstructsim.com/the-ros-development-studio-by-the-construct/>
7. Lin Q, Ke Z, Bi S, Xu S, Liang Y, Hong F, Feng L (2017) Indoor mapping using gmapping on embedded systems. In: IEEE International conference on robotics and biomimetics (ROBIO), pp 2444–2449

8. Yu N, Zhang B (2018) An improved hector SLAM algorithm based on information fusion for mobile robot. In: 2018 5th IEEE International conference on cloud computing and intelligence systems (CCIS)
9. Das S (2018) Simultaneous localization and mapping (SLAM) using RTAB-MAP. *Int J Sci Eng Res* 9(8)
10. Angeli A, Doncieux S, Meyer J, Filliat D (2008) Real-time visual loop-closure detection. In: 2008 IEEE International conference on robotics and automation
11. ROS AMCL. <http://wiki.ros.org/amcl>
12. Zheng K (2019) ROS navigation tuning guide. In: IEEE International conference on robotics and automation 8 Apr 2019
13. datknet_ros. http://wiki.ros.org/darknet_ros
14. Redmon J, Farhadi A (2018) YOLOv3: an incremental improvement. In: Computer vision and pattern recognition. IEEE, 8 Apr 2018
15. Redmon J (2017) Darknet: open source neural networks in C. <https://pjreddie.com/darknet/>

Robotic Assistant for Medicine and Food Delivery in Healthcare



Akash Bagade, Aditya Kulkarni, Prachi Nangare, Prajakta Shinde,
and Santwana Gudadhe

Abstract This paper presents the use of a three-wheel holonomic motion drive system for medicine and food delivery to patients in hospitals. The mechanical design of the holonomic drive is discussed along with the control system used for the robot. The inverse kinematic model of the three-wheel omni drive is analyzed and used for programming the navigation system. This paper analyzes the use of a gyroscope in robot heading and position control. Along with gyroscopes, the use of rotary encoders is also discussed. The encoders are used to keep track of the position of the robot while navigating. Due to the COVID-19 pandemic, mobile robots gained high demand. These robots are used to supply medicines and food to patients and staff, medical equipment required by doctors and nurses, thereby optimizing communication between doctors, hospital staff members and patients and reducing the contact between healthcare staff and patients which is useful in preventing the spread of diseases through direct contact.

Keywords Robotics · Data acquisition · Error correction · Inverse kinematics · Holonomic drive · PID controller · Automatic robot

1 Introduction

The COVID-19 pandemic impacted day-to-day life as well as healthcare services all over the world. The virus spreads by contact between the people, and that is the big issue in controlling the spread. The real problem is to stop the spread of viruses. Healthcare workers had to come in contact with the infected patients for their treatment and were getting infected by the virus. This increased the demand for the use of robots for treating infected patients.

Robots in medical healthcare help medical personnel by relieving them from routine tasks that take their time away from more important responsibilities. Robots

A. Bagade (✉) · A. Kulkarni · P. Nangare · P. Shinde · S. Gudadhe
Pimpri Chinchwad College of Engineering, Pune, India
e-mail: akashssbk@gmail.com

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
D. P. Agrawal et al. (eds.), *Cyber Security, Privacy and Networking*, Lecture Notes
in Networks and Systems 370, https://doi.org/10.1007/978-981-16-8664-1_5

can be used to deliver medicines, laboratory specimens or other sensitive material within a hospital environment and assist the medical staff ensuring faster functioning. They can be used as a mode of communication between patients with transmissible diseases and their relatives or other hospital staff. Technologically advanced countries had already started using robots, while less developed countries were struggling to get robots in healthcare. One of the major reasons behind that is the price of these robots. The high cost of these robots only makes them available to the well-developed high-income hospitals that can bear the cost, but severely limits them from helping in the hospitals across the majority of the nation who cannot afford them.

That is, when we decided to make a robot for food and medicine delivery to the patients, our intention behind this was to reduce the contact between infected patients and healthcare workers, thus ensuring the social distancing, to reduce the spread of COVID-19 and develop a cost-effective robotics solution to a problem that can be utilized by everyone.

This paper presents the use of three-wheel holonomic omni wheel drive to create a robot assistant in the field of healthcare. The robot presented is capable of delivering food and medicine to the patients without the need for healthcare workers to enter the area where patients are treated. Once given the positions where the patients are, the robot is capable of delivering food and medicines to the patients. The robot is built to fulfill the quick and short-term requirements at hospitals and is cost-effective.

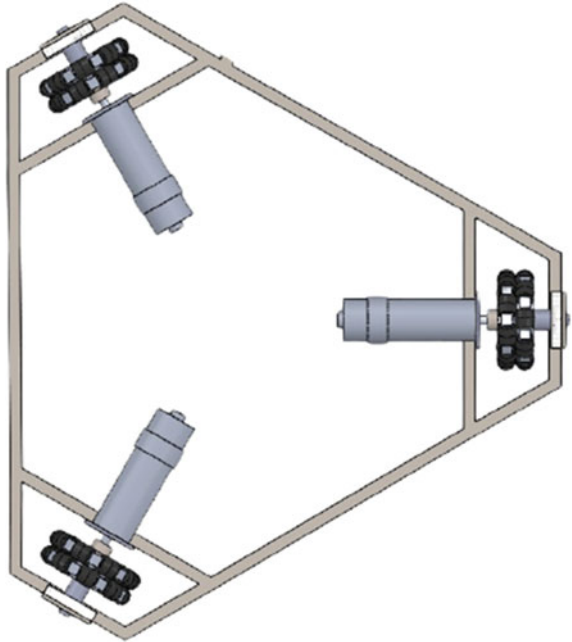
Mobile robots are broadly classified as wheeled robots and legged robots. Using wheeled robots over legged robots is more energy-efficient. Wheeled robots are further classified as holonomic (omnidirectional) and non-holonomic (non-omnidirectional). Holonomic robots have an advantage over non-holonomic that they can change the direction of motion without steering the driving wheels. Based on requirements and considering the above points, a three-wheel holonomic omni drive robot is selected.

2 The Robot

The design of the robot was made considering the hospital environments. The robot was designed to enable it to maneuver swiftly from tight places avoiding obstacles. The mechanical structure of the robot is made from stainless steel to ensure rigid structure and durability. The base three-wheel drive is hexagonal in structure with omni wheels mounted on alternate sides of the hexagon. Omni wheels of 100mm diameter driven by DC motors are used.

Planetary DC-g geared motors with a rated speed of 400 RPM and 3 Nm torque are used as driving motors. The combination of motor and wheel is selected to provide the required performance of the robot. The robot uses feedback from rotary encoders for localization and a gyroscope for error correction. The base contains a motion system (motors and wheels), battery and electronic components. Arduino Mega is used as a microcontroller.

Fig. 1 Three-wheel holonomic omni wheel drive of the robot



2.1 The Mechanical Implementation

Figure 1 shows the motion system of the robot. The robot uses three omni wheels shifted 120° from each other [6]. These three wheels are attached to three independent motors via flanges. Each wheel is at the same distance from the center of the robot.

2.2 Omnidirectional Wheels

The core of the omni wheels used is made up of aluminum plates, and the rollers are made up of plastic. The use of aluminum gives strength to wheels and is also lightweight. The wheel has 18 rollers in total. Figure 2 shows the structure of omni wheels used.

2.3 Inverse Kinematic Model

The inverse kinematics model of a three-wheel omni drive robot takes linear and angular velocities of a robot as input and produces linear velocities of individual wheels as its output.

Fig. 2 Omni wheel

Figure 3 shows the parameters of the inverse kinematic model. V is the linear velocity of the robot (in m/s), and θ is the angle of travel of the robot measured from the vertical axis (in degrees). V_a , V_b and V_c are the linear velocities (in m/s) of three wheels as shown in the diagram below. Each wheel is at some degree offset from the vertical axis which is shown in the diagram.

Figure 4 shows ω the angular velocity of the robot (in degrees/s) and R , the radius (in m).

The velocity equation of the particular wheel in terms of translational and rotational velocity of the robot is given as

$$V_i = V_{translation-i} + V_{rotation-i} \quad (1)$$

From the above equation, we can represent the velocity equation for each wheel as

$$V_a = V \cdot \cos(30 - \theta) + \omega \cdot R \quad (2)$$

$$V_b = V \cdot \cos(150 - \theta) + \omega \cdot R \quad (3)$$

$$V_c = V \cdot \cos(270 - \theta) + \omega \cdot R \quad (4)$$

where V_a , V_b and V_c represent the velocities of wheels a , b and c , as seen in Fig. 3. ω represents the angular velocity, and R represents the distance between center of the robot and the wheels as seen in Fig. 4. θ is the angle between the reference axis and the velocity vector V as seen in Fig. 3.

Fig. 3 Kinematic diagram of the motion system

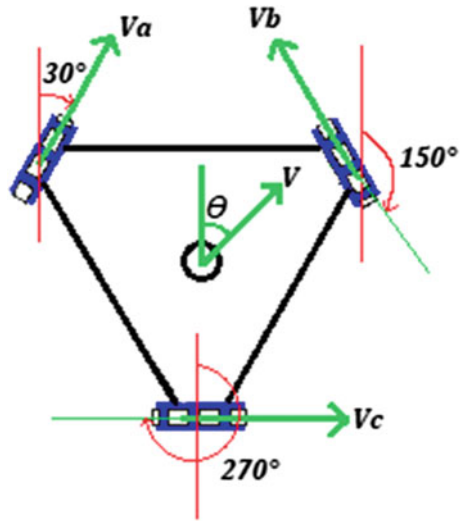
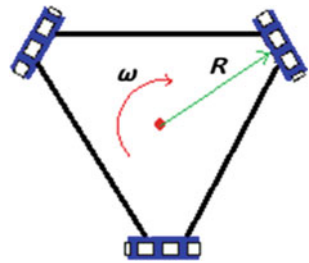


Fig. 4 Kinematic diagram showing angular velocity



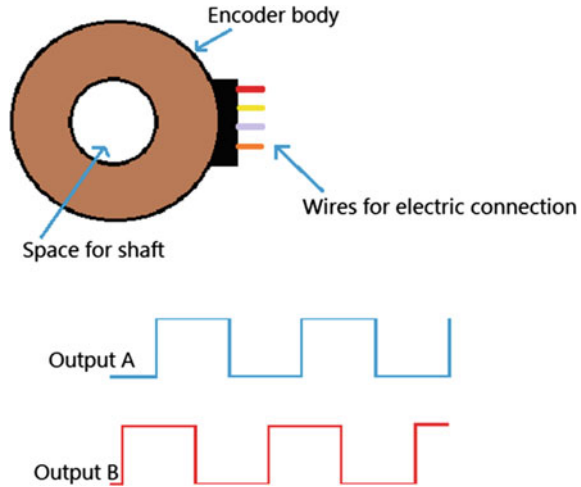
3 Control system of the robot

The navigation system of the robot is controlled by an ATmega2560-based Arduino Mega board which uses feedback from rotary encoders, proximity sensors and a gyroscope.

3.1 Rotary Encoders

For measuring the travel of the robot along the floor, we have used rotary encoders. Rotary encoders are mounted along the shafts of the wheels. They convert the angular rotation of the shaft into electric signals. These signals are then processed by the microcontroller to which they are attached. The microcontroller converts these signals into numerical values which represent the number of rotations performed by the shaft on which the encoder was mounted. Using this information, along with the

Fig. 5 Rotary encoder—incremental type



dimensions of the wheel on which these devices are mounted, we can calculate the distance traveled by the robot in any direction.

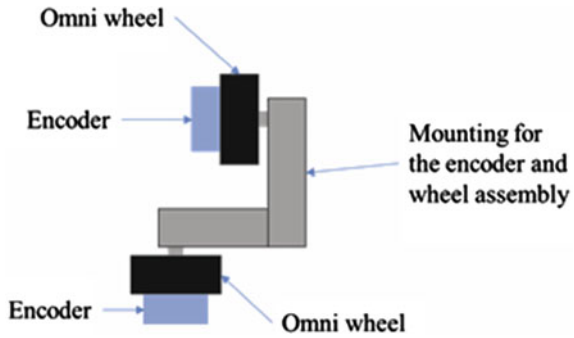
To convert signals from the encoder into distance traveled, we need to know the number of signals produced by the encoder in one rotation of the shaft (p) and the diameter of the wheel attached along the shaft along with the encoder (d in centimeters or meters). With this information, we know that for one rotation of the wheel (and the shaft), the distance traveled will be $\pi * d$ (circumference of the wheel). This is the distance traveled in one rotation of the wheel which corresponds to p signals of encoders. From this, we can conclude that, for x encoder signals, the distance traveled will be (Fig. 5):

$$D = (x/p) * (\pi * d) \quad (5)$$

D is the distance, x is the number of encoder signals, p is the number of encoder signals for one rotation, and d is the diameter of the wheel the encoder is attached to. Similarly, we can also calculate the velocity of the robot if needed by this method. Rotary encoders are of two types, namely incremental and absolute encoders. The major difference between these is that the absolute encoders are capable of retaining their position information even in case of power failure or turning the system off. Incremental encoders do not retain this information, but they are much faster and can be obtained in a wide range of precisions depending on their application. Since we are only using the encoders to determine the distance traveled, we have opted for incremental encoders.

The encoders can be mounted directly on the shafts of the driving wheels, but during our testing, we realized it is not a good way to determine the distance traveled. By mounting the encoders directly on the shaft of the driving wheels, any error in the motion of the robot such as wheels slipping on a smooth surface will be transferred

Fig. 6 Encoder mounting



over to the encoder readings. This will cause inconsistencies in the readings of the measured distance traveled.

One way to deal with this issue is to mount encoders on free rotating wheels mounted on the robot. These wheels do not have their own driving power and rotate only when the robot is in motion. Even in the case of a driving wheel slipping or skidding on the surface, the freely rotating wheels will still register close to perfect readings for distance traveled.

We have mounted two free rotating omni wheels perpendicular to each other for measuring the travel along the *X* and *Y* axes, respectively. Using both encoders together, we can determine the distance traveled by the robot in any direction in the *X-Y* plane.

These free rotating omni wheels are susceptible to vibrations at high speeds, which can induce errors in the encoder readings. During our testing, we found that for a distance of 1 meter, with an encoder of 1440 CPR, we had a maximum error in measurement of 4 mm. This error can be further reduced by providing a suspension mechanism for these wheels (Fig. 6).

3.2 Proximity Sensors

These are non-contact sensors, capable of detecting obstacles or objects placed in front of them without any physical contact with them. These are useful to detect objects around the robot quickly. These sensors work by emitting an electromagnetic field or a beam of electromagnetic radiation and observe the changes in the field or the returning signal if it hits an object (Fig. 7).

There are various types of proximity sensors available for use, like inductive, capacitive and photoelectric. Out of these, the inductive proximity sensors can only be used on objects which contain a decent amount of iron or other ferrous substances. They are not capable of detecting non-metallic obstacles. Capacitive sensors are capable of detecting both metallic and non-metallic objects, since they make use of capacitive plates, their response time is quite slow. Photoelectric proximity sensors

Fig. 7 Working of proximity sensor

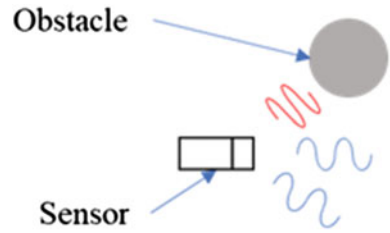
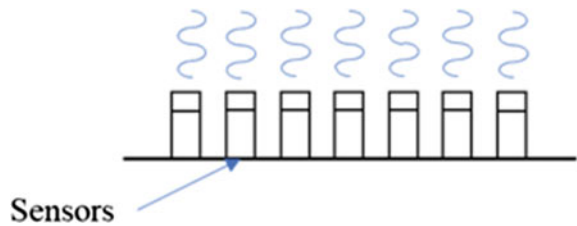


Fig. 8 Array of proximity sensors



are much better as they use a beam of light to detect the presence of an obstacle. Just like the capacitive sensors, photoelectric sensors are capable of detecting metallic and non-metallic obstacles, but they have a lot faster response time. They are capable of detecting smaller objects even at longer distances. Hence, they are the preferred mode of detecting obstacles instantly.

The proximity sensor we have used is an infrared proximity sensor that emits a beam of infrared light and detects the changes in the returning radiation of the beam. The range for detecting obstacles is around 20 cm.

Testing the sensor: The IR-based proximity sensor was able to detect a variety of objects very quickly. The only issue we faced was while detecting darker colors, mostly black. In this case, the sensor was not able to detect the obstacles at the rated range of 20 cm but was reduced to as much as 11 cm depending on the proportions of black color in the object.

We can use an array of proximity sensors along the sides of the robot to immediately detect the presence of an obstacle in any direction of the robot. Using an array of sensors will help cover the blind spots of the robot (Fig. 8).

3.3 Gyroscope

Gyroscopes are used to measure the rotation along an axis. Since the robot needs to turn to reach various destinations, a gyroscope can be used to determine how much it has turned or if it is not moving in the desired direction with the correct orientation. The robot can rotate only around the Z axis, and we will only be using the Z-axis rotational values.

While traveling along a specified direction, the robot needs to maintain its heading. We can verify if the robot is maintaining its heading using the gyroscope. The gyroscope will provide us with the robot’s current angle which can be compared with the desired heading angle, and depending on the values, we can apply a suitable correction algorithm such as a PID controller to ensure that the robot maintains a correct heading.

The PID controller for the gyroscope can be applied as follows:

$$\text{correction} = (k_p * \text{error}) + (k_i * \text{total_error}) + (k_d * \text{change_in_error}) \quad (6)$$

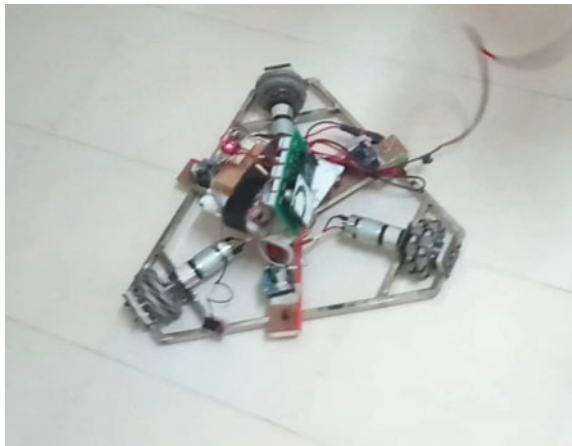
where the error is the difference between the current angle and the desired angle, total_error is the accumulated error over time, and change_in_error is the difference between currently calculated error and previously calculated error. k_p , k_i and k_d are the constants for the proportional, integral and derivative terms of the equation, respectively. This is a simple implementation of the PID controller with change in time dt considered as constant for integral and derivative calculations.

Depending on the gyroscope module used, the readings from the sensor can have varying levels of noise. In such a case, noise can be reduced by the use of algorithms such as the Kalman filter.

4 Testing of the Robot

The robot drive was tested to travel to a specified position taking feedback from rotary encoders and maintaining position by taking feedback from the gyroscope (Fig. 9).

Fig. 9 Drive testing



Test Scenario 1: Adjusting the PID controller for error correction. Result: Success. The robot uses a gyroscope for maintaining its angular orientation. The PID controller was tuned by manually introducing an error in angular position and observing the responsiveness of the robot. In case of slower response, the constants k_p and k_i were increased. To reduce overshoot, constant k_d was increased.

Test Scenario 2: Traveling a predefined path. Result: Success. A path is a collection of movements in straight lines. The robot was able to detect its position and travel along the given path using feedback from the rotary encoders and gyroscope.

Test Scenario 3: Traveling on a path with obstacles. Result: Success. The robot was made to travel on a path, and both stationary (boxes, tables) and dynamic (people) objects were introduced in its path. The robot was able to detect all the obstacles within a 20 cm range and stop its motion before making contact with any of them.

Test Scenario 4: Traveling on a path with obstacles and introducing error in its motion. Result: Success. As an extension to Test 4, deflection in the robots path was purposefully introduced to observe its actions. The robot was able to successfully correct its position and start following its defined path.

5 Future work

The robot is using an encoder for localization. The future aim would be to use a camera and lidar-based localization systems for robots so as to make them more robust. Also, we would look to attach a sanitization system on robots as well as video call support.

6 Conclusions

This paper presented the use of a three-wheel holonomic motion drive system implemented for medicine and food delivery to patients in hospitals. The paper describes the mechanical implementation, inverse kinematics and control system of the robot. The robot was able to follow a given path taking feedback from encoders and gyroscopes.

References

1. A review on implementation of robotic assistance in COVID-19 epidemics. *Int J Adv Sci Technol* (2020)
2. Robotics utilization for healthcare digitization in global COVID-19 management. *Inte J Environ Res Publ Health*
3. Vehicle routing problem with pickup and delivery of multiple robots for hospital logistics (2016)
4. Healthcare robot systems for hospital environment: CareBot and ReceptionBot (2015)
5. Designing a robotic assistant for healthcare applications (2009)
6. Design, implementation and validation of the three wheel holonomic motion system of the Assistance Personal Robot(APR) (2016)

Privacy-Preserving Record Linkage with Block-Chains



Apoorva Jain  and Nisheeth Srivastava 

Abstract We present a method for privacy-preserving record linkage of partially de-identified data. The key novelty of our proposed method is the use of a block-chain to store user information, preventing privacy leaks even if all system databases and cryptographic protocols have been compromised by an adversary, including databases containing identifying information. While satisfying this stringent privacy constraint, the system remains able to deterministically link and retrieve de-identified records to unique identities. With logarithmic time complexity to perform linkage and retrieval operations, our method is highly scalable for large-scale use cases. Designed to be HIPAA (Health Insurance Portability and Accountability Act) compliant by default, such systems are easily adaptable to large real-world healthcare systems.

Keywords De-identification · Privacy-preserving record linkage (pprl) · Block-chain · Identity management

1 Introduction

It is at this point a cliché to say that we live in an age of extensive and ubiquitous data mining. In response to privacy concerns surrounding such activities, privacy-preserving data linkage methods are being extensively studied [1]. Blind data linkage or privacy-preserving record linkage (henceforth, PPRL in this paper) is defined as the process of finding records that represent the same individual without revealing the identity of the individuals.

Existing PPRL methods use a combination of secure communication and distributed system architectures to ensure privacy while merging data. For instance, it is common practice to store a mapping from identifying information to a hashed token securely, and use the token to index records for data mining and other uses.

A. Jain (✉) · N. Srivastava
Department of CSE, IIT Kanpur, Kanpur, India
e-mail: apoorvaj@cse.iitk.ac.in

N. Srivastava
e-mail: nsrivast@cse.iitk.ac.in

Such systems help reduce privacy threats, but are not foolproof [2]. In particular, it is frequently possible to infer peoples' identities based on analysis of side information available in the data. For instance, the identity of a person suffering from a rare disease can be easily guessed from an EHR database, and the identity of a family that has recently moved into a neighborhood can easily be guessed from a property tax database.

Hardening external network communication via the use of SSL and other cyber-secure practices can protect against the possibility of privacy leaks from malicious external actors to a considerable degree. However, the risk of such leaks from insider attacks cannot be handled in the same way [3]. As our economic lives increasingly move online, the opportunity cost of privacy leaks continues to grow in a direction incentivizing insider threats for most businesses that collect sensitive user information.

In this paper, we propose the use of block-chains to construct a PPRL system resistant to both external and insider privacy threats. We have developed this system for a specific use case—ensuring privacy for a platform for conducting behavioral experiments, but the approach used can easily generalize to simpler use cases of practical import.

The specific use case we have designed our system for requires collection of data from different participants over a period of months over multiple sessions. This includes behavioral data including their habits, health, social interactions, overall economic transactions, etc. The platform permits payments to participants based on valid responses. As the project aims to gather extremely minute details about the daily lives of the participants, privacy and security is a dominant priority. The data should be secured from any external as well as internal adversary attacks. At the same time, the need to make payments requires that sensitive identifying information must be collected and stored, so complete de-identification is not feasible.

We follow HIPAA compliant system design by de-identifying the data before storing it inside the database that research teams would access [4]. We ensure that stored session data does not contain any personally identifier directly or indirectly (quasi identifiers i.e. a combination of partial identifiers like year of birth, gender, pin code, etc.).

But this de-identified data needs to be correctly linked with earlier de-identified submissions of the same user so that it can be used for research purposes. Thus, we need to perform privacy-preserving record linkage within de-identified experimental behavioral records with security against any kind of internal as well as external adversary attacks. This could be achieved with appropriately selected hashing schemes, as discussed below in related work.

However, while linking records, we also need to maintain deterministic linkage quality between response data and actual real identities in terms of both sensitivity (correctly linking together all records of the same person) and specificity (avoiding false records linking of different people) to prevent incorrect payments. So, we ultimately require privacy-preserving data linkage of de-identified behavioral records collected and stored within a single database. In order to support payments, we also require to relate these linked records with the separate database containing actual

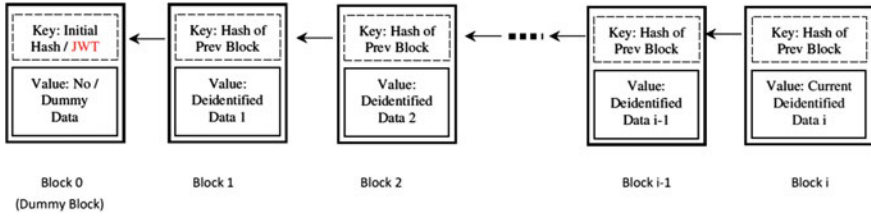


Fig. 1 Block-chain of a de-identified user depicting how de-identified data submissions are linked together

identifying information. The latter linking with actual identities should be done in-memory at run time, and the linkage results should not be stored anywhere (except the calculated payment amount) in order to support payment operations and preserve privacy of the users without leaking any information to adversaries outside or inside the system.

To satisfy the stringent privacy constraint and prevent incorrect payments, we have devised a block-chain-based method which uses one block-chain per user as shown in Fig. 1. Each de-identified session data of a user is stored in separate blocks of the block-chain, and all the blocks across different block-chains of all the users are mixed randomly leading to a single collection of blocks inside the database.

2 Related Work

Privacy-preserving record linkage is performed in a wide variety of applications [5]. PPRL techniques based on Bloom filter (BF) [6] are highly used in many real-world applications such as healthcare as these provide good linkage quality. Moreover, for large databases, BF-based methods are easily scalable [7]. But recent studies [8], [9] have shown that Bloom filter-based encodings are open to various cryptanalysis attacks. Sensitive attributes can be re-identified from BFs using the recurrence counts and bit patterns in BFs and public list of the most frequent attributes easily available. Similarly, hash (over personal identifiers)-based methods are prone to dictionary and frequency attacks [5]. Also, in practical settings, these sometimes fail to provide good linkage quality (sensitivity) without extensive pre-processing because of the nature of hash encoding which totally changes even if only a single character is changed in the encoding string.

Moreover, both BF and hash-based PPRL techniques use personal identifiers from each record to link the data records [5]. But in order to achieve better privacy and security, we would like to use data de-identified at the source, i.e., removing all personally identifying attributes from the records at the client side itself.

Nguyen et al. [10] proposed a solution to a very similar problem in healthcare settings where they need to link de-identified data of individual patients collected

from multiple healthcare sites over a period of time. Their system has multiple source points (medical labs) to collect data for a particular user over a period of time whereas we have a centralized/single source point (web app) with a unique account for each user. In terms of approach, they generated irreversible hashed linkage keys based on patient-identifying data captured in the patient electronic medical records (EMRs) at the site followed by removing these patient identifiers, i.e., making the data de-identified. For linking all these de-identified records of a patient captured over multiple sites, they stored it with these common hashed linkage keys inside the database. Based upon these common linkage keys, all the records of a patient are linked together.

However, in the above approach, an insider can easily link all the records of a patient using common linkage keys by just looking at the database. Hence, it's vulnerable to insider attack. We are looking for a method that is resistant to insider attack.

3 Proposed Approach

In this section, we discuss our block-chain-based approach which is robust to database-targeted insider attacks, viz. when the adversary is able to read or copy from the system's databases but is unable to run system operations. Unlike existing PPRL solutions, our method is robust to insider attack threats, as we describe below.

In Fig. 2, table (a) displays records stored using hash-based approach, whereas table (b) displays records stored using our block-chain-based approach. De-identified Record a and Record d of both tables belongs to a single user. By just looking at table (a) of Fig. 2, anyone can easily observe that record a and record d has a common key, so it belongs to the same user. But as we can see in table (b), key of Record d is different from key of Record a. The key is equal to the hash of whole block containing

Hash based Linkage Key	Data	Blockchain Based Key	Data
78ae570d7e0d4184f0b3 b65970cd886c5cfb0c6d	Record a	78ae570d7e0d4184f0b3 b65970cd886c5cfb0c6d	Record a
dcf30ba345198e8e329a da19a7d35bc122e84ae8	Record b	dcf30ba345198e8e329a da19a7d35bc122e84ae8	Record b
14412968add99f8b23b5 7dc9392c921326af61d9	Record c	b933469b1364d9dd8be5 6f56bcc10230c1258778	Record c
78ae570d7e0d4184f0b3 b65970cd886c5cfb0c6d	Record d	14412968add99f8b23b5 7dc9392c921326af61d9	Record d

(a)
(b)

Fig. 2 Difference in how de-identified data records are stored for linkage **a** using hash-based method, **b** using block-chain-based method

previous record *a* as discussed in Sect. 3.1. So, in the latter case, by looking at the database, it is impossible for any internal adversary who has access to database of records to link all the records of any user.

Even if we modify the above hash-based system to generate unique hashes for each of the Records *a* and *d* by using some cryptographic salt-based mechanisms and separately keep a record of those salts, still the hash-based approach would be vulnerable to write access insider attacks. If some internal actor with write access wants to falsify information by manipulating data, hash-based approach remains repudiable.

In contrast, in a block-chain-based method some records would be left unlinked to any of the block-chain as hash of previous block would change with manipulation in data. This would signal that some data manipulation attempts have been made and no false information would be shared with the researchers. Backups could be used to restore data to its original state.

3.1 *Privacy-Preserving Record Linkage*

We use a block-chain—a linked list built with hash pointers—as a data structure to store and link the experimental submission data of each user inside the centralized NoSQL database [11].

We use this structure to store experimental data of each user in a separate block-chain where key to current block is the hash pointer of the previous block in the block-chain of the current user as shown in Fig. 1. Blocks across different block-chains of all the users are mixed randomly leading to a collection of blocks inside the database. When a new user registers, a dummy block—Block 0 is generated with random hash as key and dummy data as value. This block is added in the collection of blocks of all the users. Later, when some *i*th submission by the same user arrives for storage along with the JSON Web Token (JWT) (which is equal to random hash stored in Block 0, as discussed in Sect. 3.2), key for Block 1 can be easily calculated by locating Block 0 with the help of JWT value. The block-chain is traversed till we find the key for the *i*th block corresponding to which *i*th submission data is stored.

3.2 *Partial De-identification at Source*

When a user successfully logs into the system, server sends JWT to the user as HTTP only cookie. User submits the de-identified experimental data response (which does not contain any identifier) w.r.t this JWT token to the server. This token is nothing but initial random key hash which points to Block 0, i.e., dummy starting block of the block-chain of this particular user.

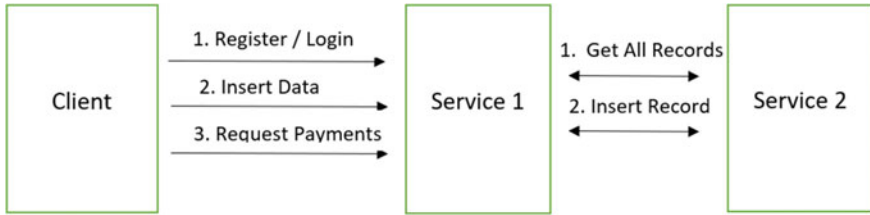


Fig. 3 System architecture

Next, block-chain is traversed to find correct key for the incoming data. As this key is the hash of the last submission block of this user, current record is correctly linked with the block-chain of this user among the collection of blocks of all the users.

4 System Design

Figure 3 shows the architecture of the system. Client may send the request to register a new user/login, insert data, request payments. These requests are received by Service 1. Service 2 is abstracted from the client and is visible to only Service 1. It accepts post requests to insert records and get request to return all records.

4.1 Service 1

Service 1 handles user profile management, using OTP verification of government IDs to guard against sybil attacks on the system. A user record consists of 'initial-Block' as a field as shown in Fig. 4. This field is initialized with a random unique hash value for each user. The random hash value acts as key for Block 0 corresponding to start of the block-chain of a user. This is similar to dummy pointer value used in linked list to point the head of linked list. This random hash value along with some dummy data is also inserted as a block in the experimental record database via Service 2 as shown in Fig. 5a.

When a registered user sends a login request along with correct login credentials, a JWT token with payload equal to the 'initialBlock' value is returned back to the user.

Next, the session's response data submitted by a registered user along with the JWT is received by Service 1 to be inserted as experimental record. All the sorted experimental records are fetched from Service 2. Binary Search is performed on all these records to find a key with the corresponding JWT token value. Using this, we calculate the key for the next block which is equal to hash of the found record block and traverse the block-chain. We repeat this procedure until we reach the end of the

```

_id: ObjectId("6081868d9a90c617b46cdf63")
Govt ID: "12345678"
fullName: "Alice"
created: 2021-04-22T14:22:05.895+00:00
initialBlock: "78ae570d7e0d4184f0b3b65970cd886c5cfb0c6d"
currentBlock: "78ae570d7e0d4184f0b3b65970cd886c5cfb0c6d"
__v: 0

```

Fig. 4 Data in user records database at Service 1 with initialBlock initialized with random hash value

<pre> _id: ObjectId("6081868e665a9057a0f071b2") value: Array 0: "dummy data" key: "78ae570d7e0d4184f0b3b65970cd886c5cfb0c6d" created: 2021-04-22T14:22:06.660+00:00 __v: 0 </pre> <p style="text-align: center;">(a)</p>	<pre> _id: ObjectId("608187ac665a9057a0f071b3") value: Array 0: Object 1: Object key: "14412968add99f8b23b57dc9392c921326af61d9" created: 2021-04-22T14:26:52.262+00:00 __v: 0 </pre> <p style="text-align: center;">(b)</p>
--	--

Fig. 5 Data at experimental records database at Service 2 **a** Block 0 with key= JWT/ initialBlock and value = dummy data, **b** Block 1 with key = $h(\text{Block } 0)$ and value = De-identified Data

block-chain. Finally, new data is inserted in experimental records database with key as hash of the last block in this block-chain as shown in Fig. 5b. This can be seen as appending block at the end of the block-chain.

Finally, the user may request for payments based on amount of experimental data submitted by her. This request is received by Service 1. This is followed by calculating payments for this user and updating this amount in internal records to be paid in next month’s payment cycle.

Payments are calculated in a similar way to how data is appended in the block-chain by traversing the whole block-chain. While traversing the block-chain, we note the amount of data in each block, filtering out payments to incomplete, unrealistic and/or spoofed responses after which payments are calculated. A dummy record reflecting payments success is appended at the end of the block-chain. The key of this dummy record acts as new ‘currentBlock’ value for this user and is updated in the user record. This ensures that in future payments are calculated considering the data blocks after this point only.

4.2 Service 2

This service is a simple wrapper around the application’s database. It accepts POST requests to insert a block inside the Experimental Record database. It also accepts a GET request to return all data blocks sorted by key. A response consisting of all sorted blocks is returned as a JSON object.

5 Performance Analysis

We contrast our solution with the solution proposed by Nguyen et al. [10] to a very similar problem. In a larger gold-standard dataset with a lower level of data completeness, their solution gave specificity value ranging between 91% and 99% while the sensitivity range is 94–95%. In contrast, because we retain real identities within our system, our method offers deterministic linkages.

Looking for a particular key among a pool of keys takes linear time using linear search. We have sorted this key-value pair blocks collection and used binary search to reduce searching time to $O(\log n)$. While traversing the block-chain, searching is followed by calculating hash over the current block that takes $O(L)$ time, where L is the length of the block. The $O(L)$ time is generally considered constant $O(1)$ for comparatively smaller size blocks. As number of data blocks (experimental sessions) per user $m \ll n$ (size of database) and length of each block $L \ll n$, block-chain traversal and linkage take $O(m(\log n + O(L))) \approx O(\log n)$. Thus, our solution is scalable to large scale systems.

6 Security Analysis

De-identifying data and storing it inside blocks of a block-chain preserves privacy of the users under normal operation of data analytic activities in the system. If an adversary gains read access to both the databases—the first comprising of real identity of the users with JWT token value and the second comprising of experimental data submissions by all of them, the adversary still cannot map what data belongs to which user until and unless the adversary knows the complete block-chain algorithm and is able to execute it on the database. This is because in the second database, dummy data is stored with respect to the JWT (Block 0). Real experimental data starts from Block 1 which is not mapped with JWT anywhere in the second database. By just looking at both the databases or running search algorithms, it is impossible to map real experimental data with real identities. Even if an adversary learns the block-chain traversal algorithm but has only access to the database 2, i.e., experimental records database (or a copy of the database via a data dump), identifying information can still not be extracted. Thus, our system can retain privacy of the users even in the extreme circumstance of a threat vector gaining read access to all the relevant databases caused by external as well as insider adversary.

Next, as the key for the next block is calculated dynamically as a hash over the whole previous block (and not over static specific attributes), our method will be relatively robust to dictionary/frequency attacks. Crucially, the dynamic key for each of the submitted record of a user makes it impossible for an adversary with read rights to do record linkage among random collection of records. Thus, our block-chain-based method works against database-targeted insider attacks that may leak private

linkage information as compared to pre-existing state-of-the-art PPRL methods [10] which are vulnerable to such kind of insider attacks.

Further, as discussed in Sect. 3, our method is relatively resistant to data tampering attacks associated with write access insider threats.

7 Conclusion

We have developed a system that partially de-identifies data at source while permitting deterministic linkage of de-identified data to real identities. The use of a block-chain in storing and retrieving data ensures that user privacy can be retained even in the extreme circumstance of a threat vector gaining read access to all relevant databases. This additional facet of privacy protection makes our solution robust to the most common form of insider attacks—unauthorized data dumps to external sources [12]. Though we have designed our approach to be HIPAA compliant, partial de-identification and deterministic linkages provided by our approach gives the flexibility to an enterprise to delete the participant’s data if any participant chooses to delete that in future as per GDPR compliance—right to be forgotten. Preserving privacy even in the face of complete disclosure of data also opens up the possibility of easier data sharing across enterprises without having to resort to federated learning schemes, which generally under-perform centralized analysis [13].

The use of block-chains in privacy-preserving data storage has been anticipated in recent literature on the subject. For example, Miyachi & MacKey extol the benefits of incorporating block-chain-based computations to design privacy-preserving health information systems in a recent concept note [14]. Similar frameworks have been recently proposed for managing IoT data [15]. Our solution provides a concrete instantiation of PPRL that could potentially be used as a building block for such frameworks. Block-chains have also been used recently to develop a privacy-preserving contact tracing system for the ongoing Covid-19 pandemic [16]. This solution resembles ours significantly in combining a device-specific prefix with location suffixes to track users’ location in a privacy-preserving manner. However, since their system uses a human in the loop—a trusted diagnostician—to map device-based pseudonymous suffixes with real identities, and thus is not a direct comparison for our system, wherein real identities reside in a centralized system, but are still robust to privacy leaks.

While our system is designed as a solution for a specific problem, this general approach, as also exemplified in [16], can be extended to several other differential privacy use cases. For instance, it is easy to imagine healthcare prognostics services paying people for using their personal data for both prognosticating their personal health, and simultaneously improving the generalizability of machine learning models. Differential privacy approaches such as ours could enable users to own and monetize their data at will, as conceived in recent theoretical proposals [17], without incurring the overheads of federated learning.

References

1. Agrawal R, Srikant R (2000) Privacy-preserving data mining. In: Proceedings of the 2000 ACM SIGMOD international conference on management of data, pp 439–450
2. Langarizadeh M, Orooji A, Sheikhtaheri A, Hayn D (2018) Effectiveness of anonymization methods in preserving patients' privacy: a systematic literature review. *eHealth* 80–87
3. Pfleeger CP (2008) Reflections on the insider threat. In: *Insider attack and cyber security*. Springer, Boston, pp 5–16
4. Ness RB, Committee Joint Policy (2007) Influence of the HIPAA privacy rule on health research. *Jama* 298(18):2164–2170
5. Vatsalan D, Christen P, Verykios VS (2013) A taxonomy of privacy-preserving record linkage techniques. *Inf Syst* 38(6):946–969. ISSN 0306-4379. <https://doi.org/10.1016/j.is.2012.11.005>. <http://www.sciencedirect.com/science/article/pii/S030643791200147>
6. Schnell R, Bachteler T, Reiher J (2009) Privacy-preserving record linkage using Bloom filters. *BMC Med Inform Decis Mak* 9:41. <https://doi.org/10.1186/1472-6947-9-41>
7. Randall SM, Ferrante AM, Boyd JH, Bauer JK, Semmens JB (2014) Privacy-preserving record linkage on large real world datasets. *J Biomed Inf* 50:205–212. ISSN 1532-0464
8. Niedermeyer F, Steinmetzer S, Kroll M, Schnell R (2014) Cryptanalysis of basic Bloom filters used for privacy preserving record linkage. *J Privacy Confidential* 6(2). <https://doi.org/10.29012/jpc.v6i2.640>
9. Christen P, Ranbaduge T, Vatsalan D, Schnell R (2019) Precise and fast cryptanalysis for bloom filter based privacy-preserving record linkage. *IEEE Trans Knowl Data Eng* 31(11):2164–2177. ISSN 1558-2191. <https://doi.org/10.1109/TKDE.2018.2874004>
10. Nguyen L, Stoové M, Boyle D, Callander D, McManus H, Asselin J, Guy R, Donovan B, Hellard M, El-Hayek C (2020) Privacy-preserving record linkage of deidentified records within a public health surveillance system: evaluation study. *J Med Internet Res* 22(6):e16757. <https://www.jmir.org/2020/6/e16757>. <https://doi.org/10.2196/16757>
11. Narayanan A, Bonneau J, Felten E, Miller A, Goldfeder S (2016) *Bitcoin and cryptocurrency technologies: a comprehensive introduction*. Princeton University Press, USA. ISBN 0691171696
12. Greitzer FL, Moore AP, Cappelli DM, Andrews DH, Carroll LA, Hull TD (2008) Combating the insider cyber threat. *IEEE Secur Privacy* 6(1):61–64
13. Wei K, Li J, Ding M, Ma C, Yang HH, Farokhi F, Poor HV (2020) Federated learning with differential privacy: algorithms and performance analysis. *IEEE Trans Inf Forensics Secur* 15:3454–3469
14. Miyachi K, Mackey TK (2021) hOCBS: a privacy-preserving blockchain framework for healthcare data leveraging an on-chain and off-chain system design. *Inf Process Manag* 58(3):102535
15. Daghmehchi Firoozjahi M, Ghorbani A, Kim H, Song J (2020) Hy-Bbidge: a hybrid blockchain for privacy-preserving and trustful energy transactions in Internet-of-Things platforms. *Sensors* 20(3):928
16. Xu H, Zhang L, Onireti O, Fang Y, Buchanan WJ, Imran MA (2020) BeepTrace: blockchain-enabled privacy-preserving contact tracing for COVID-19 pandemic and beyond. *IEEE Internet Things J*
17. Mills S (2019) Who owns the future? Data trusts, data commons, and the future of data ownership. In: *Data trusts, data commons, and the future of data ownership*, 4 Sept 2019

Performance Analysis of Rectangular QAM Schemes Over Various Fading Channels



Siddhant Bhatnagar, Shivangi Shah, and Rachna Sharma

Abstract There is an increased need for higher data rates within the bounds of existing bandwidth. Quadrature amplitude modulation (QAM) is a much-favored modulation scheme used in wireless communications due to its ability to achieve a high data rate while utilizing existing bandwidth. Especially, the rectangular QAM (RQAM) has received more attention due to its tactical advantages and its usefulness across various communication models. However, as the data rate is increased, the signal becomes more prone to error in the channel. There are several fading channel models used to represent channel conditions in various types of wireless communication links. Here, the radio frequency communication (RF) and optical communication links are taken into consideration. In RF, the Rayleigh, Rician, and Nakagami-m fading models are widely used models and considered in this paper. In optical communication, the log-normal fading model, which is best suited to represent weaker turbulence conditions in the channel, is considered. In this paper, the error probability of the rectangular QAM modulation scheme, subject to the fading channels of RF and optical communication, is analyzed and derived. The results are verified in Python.

Keywords Line of Sight (LoS) · Non-Line of Sight (NLoS) · Rectangular quadrature modulation (RQAM) · Square quadrature amplitude modulation (SQAM) · Bit error rate (BER) · Bit error probability (BEP) · Symbol error probability (SEP) · Signal-to-noise ratio in db (SNR)

1 Introduction

A wireless communication link faces harsh environments resulting in the achieved spectral efficiency being lower than the theoretical limits. There are more applications and devices being connected to the network through wireless links, and hence, the reliance on such links has also increased. This also means that there is increased

S. Bhatnagar (✉) · S. Shah · R. Sharma
Institute of Technology, Nirma University, Ahmedabad, GJ, India
e-mail: 17bec151@nirmauni.ac.in

traffic of data. With the increasing demand for data, the requirement of bandwidth is also to be increased. The performance of high data rate links with limited bandwidth have also been a concern. With systems having low latency, high data rate, and high reliability becoming more common, the task of modeling accurate communication models becomes more substantial. This can be through using a specific modulation scheme such as quadrature amplitude modulation and analyzing various fading channel models which are used to represent the channel conditions

There are various sub-domains of wireless communications, such as RF communication, microwave communication, and optical communication. In this paper, we have considered RF communication and optical communication. For RF communication, there are several use cases such as mobile communication, television broadcasting, and radar systems. There are many channel models used to represent the fading in such links. The Rayleigh fading channel model is best suited for non-Line of Sight (NLOS) links, while the Rician fading channel model is suited where there is a Line of Sight (LOS) path available. The Nakagami- m fading model is rather empirical and does not exist phenomenologically. The m parameter is the fading parameter and ranges from 0.5 to infinite, defining the severity of fading [1]. For optical communication, there are applications in vehicular communication, underwater communication, Li-Fi, etc. In optical channels, the phenomenon of the scintillation effect causes turbulence varying from weak to strong and can be measured by means of Rytov parameter. The log-normal fading model is used to represent the weak turbulence conditions in an optical communication channel [2].

While increasing the data rates, the scarcity of bandwidth has to be considered. The quadrature amplitude modulation is a spectrally efficient scheme that enables a high data rate without increasing the bandwidth. The QAM schemes have earned great attention in overcoming destructive channel impairments. Especially, the RQAM scheme has been able to achieve reliable links and improved voice and data throughput. This particular modulation scheme has found its usefulness where the environmental conditions in the wireless links are quite harsh or when there is a requirement of unequal protection from error or more robust protection for selective information bits [3].

The performance analysis of RQAM schemes for various fading channels has been evaluated in [4, 5]. In [4], the closed-form expression of bit error probability is derived for the RQAM over Rayleigh fading channel. In [5], the error performance is analyzed for RQAM over the Rician fading channel. The symbol error rates for RQAM over Nakagami- m fading channel and log-normal fading model are derived in [3, 6], respectively.

In this paper, we have presented error probability analysis for the RQAM modulation scheme subject to fading channels. In RF, we have analyzed the error probabilities for RQAM scheme subject to the Rayleigh, Rician, and Nakagami- m fading channels. We have also derived error probability for RQAM subject to the log-normal fading channel.

The rest of the paper is organized as follows: In Sect. 2, the rectangular QAM modulation scheme is discussed. In Sect. 3, we have detailed the bit error probability and symbol error probability for various RF fading channels and derived the symbol error probability for log-normal fading used in optical communication. In Sect. 4, we have presented the results, and lastly in Sect. 5, we have concluded our work.

2 Rectangular Quadrature Amplitude Modulation

RQAM scheme for 4×2 , 8×2 and 16×1 is shown in Figs. 1, 2 and 3. Here, we have used multiple RQAM constellations, which are generally used in real case scenarios. It is generic nature consisting of an ASK and PSK combination which allows the user to implement the scheme in several formats of QAM and non-QAM schemes as well. As it offers flexibility when it comes to the size and shape, with independently controlled distance within the in-phase and quadrature-phase components, this scheme also becomes favorable for optical communication due to this nature of maneuverability. There is an increased interest in this modulation scheme in the several applications such as vehicle-to-vehicle communication and underwater communication as well [6, 7].

3 Error Probability Analysis for RQAM Over Fading Channels

The general-order RQAM BEP for AWGN channel can be attained by [2].

Fig. 1 4×2 RQAM

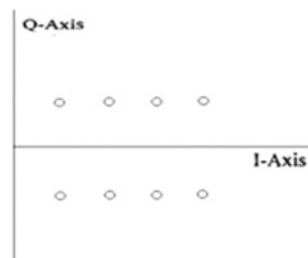
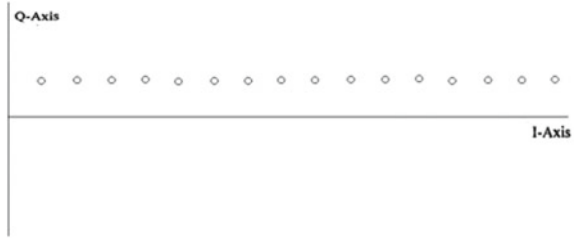


Fig. 2 8×2 RQAM



Fig. 3 16×1 RQAM

For in-phase and quadrature-phase components

$$P_I(k) = \frac{1}{I} \sum_{i=0}^{(I-2^{-k})I-1} \left\{ \Theta(i, k, I) \times \operatorname{erfc} \left(\frac{(2i+1)d_i}{\sqrt{N_0}} \right) \right\} \quad (1)$$

$$P_J(l) = \frac{1}{J} \sum_{j=0}^{(J-2^{-l})J-1} \left\{ \Theta(j, l, J) \times \operatorname{erfc} \left(\frac{(2j+1)d_j}{\sqrt{N_0}} \right) \right\} \quad (2)$$

where

$$\Theta(a, b, c) = (-1)^{F\left[\frac{a \times 2^{b-1}}{c}\right]} \times \left[2^{b-1} - F\left(\frac{a \times 2^{b-1}}{c} + \frac{1}{2}\right) \right]; \quad (3)$$

and $F(x)$ is the floor function. The general-order RQAM ASEP, P_s , is attained only with the averaging error probability that is [7]

$$P_s = \int_0^{\infty} P_e(\gamma) f_\gamma(\gamma) d\gamma \quad (4)$$

where $P_e(\gamma)$ is given by

$$P_e(\gamma) = 2 \left(1 - \frac{1}{M_I} \right) Q(A_I \sqrt{\gamma}) + 2 \left(1 - \frac{1}{M_Q} \right) Q(A_Q \sqrt{\gamma}) - 4 \left(1 - \frac{1}{M_I} \right) \left(1 - \frac{1}{M_Q} \right) Q(A_I \sqrt{\gamma}) Q(A_Q \sqrt{\gamma}) \quad (5)$$

$$A_I = \sqrt{6 / [(M_I^2 - 1) + r^2(M_Q^2 - 1)]} \quad (6)$$

and

$$A_Q = \sqrt{6r^2 / [(M_I^2 - 1) + r^2(M_Q^2 - 1)]} \quad (7)$$

where $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{t^2}{2}} dt$ and M_I and M_Q represent the in-phase and quadrature-phase signal and γ is signal-to-noise ratio.

3.1 Rayleigh Fading Model

Rayleigh fading channels are suited when there are a lot of objects in the proximity environment. Thus, this channel is used for different kinds of scenarios including city infrastructure.

$$P_I(k) = \frac{1}{I} \sum_{i=0}^{(1-2^{-k})I-1} \left\{ w(i, k, I) \times \left(1 - \frac{\sqrt{\frac{3(2i+1)^2 \log_2(I \times J) \times \gamma}{I^2 + J^2 - 2}}}{\sqrt{\frac{3(2i+1)^2 \log_2(I \times J) \times \gamma}{I^2 + J^2 - 2} + 1}} \right) \right\} \quad (8)$$

$$P_J(l) = \frac{1}{J} \sum_{j=0}^{(1-2^{-l})J-1} \left\{ w(j, l, J) \times \left(1 - \frac{\sqrt{\frac{3(2j+1)^2 \log_2(I \times J) \times \gamma}{I^2 + J^2 - 2}}}{\sqrt{\frac{3(2j+1)^2 \log_2(I \times J) \times \gamma}{I^2 + J^2 - 2} + 1}} \right) \right\} \quad (9)$$

with

$$w(i, k, I) = (-1)^{\lfloor \frac{i \times 2^{k-1}}{I} \rfloor} \times \left(2^{k-1} - \left[\frac{i \times 2^{k-1}}{I} + \frac{1}{2} \right] \right) \quad (10)$$

$$w(j, l, J) = (-1)^{\lfloor \frac{j \times 2^{l-1}}{J} \rfloor} \times \left(2^{l-1} - \left[\frac{j \times 2^{l-1}}{J} + \frac{1}{2} \right] \right) \quad (11)$$

The equation between distance d_j and d_i , energy E_b in RQAM $I \times J$ system is

$$E_b = \frac{(I^2 - 1)d_i^2 + (J^2 - 1)d_j^2}{3 \log_2(I \times J)} \quad (12)$$

For Rayleigh channel, the bit error probability of $I \times J$ RQAM is

$$P_{b, \text{Ray}} = \frac{1}{\log_2(I \times J)} \left(\sum_{k=1}^{\log_2(I)} P_I(k) + \sum_{l=1}^{\log_2(J)} P_J(l) \right) \quad (13)$$

3.2 Rician Fading Model

This fading occurs whenever one of the paths like Line of Sight signal or other strong reflection signals is stronger than others. TWDP that is the two waves with diffuse

power is a special case of Rician fading channel. This can be statistically described as a process with nonzero mean, and hence, the envelope will follow the Rician distribution.

The bit error probability for the Rician model is derived for diversity in [4] and can be stated for without diversity as

$$P_{b, \text{text Ric}} = \frac{1}{P\pi} \sum_{j=0}^{(i-2^{-i})P-1} (-1)^{\lfloor \frac{j2^{i-1}}{P} \rfloor} \times \left(2^{i-1} - \left[\frac{j \times 2^{i-1}}{P} + \frac{1}{2} \right] \right) \times \int_0^\pi \frac{\exp\left[-\frac{K \frac{3(2j+1)^2 \csc^2 \theta}{(I^2+J^2-2)}}{\frac{K}{E} + \frac{3(2j+1)^2 \csc^2 \theta}{(I^2+J^2-2)}}\right]}{\left(\frac{K}{E} + \frac{3(2j+1)^2 \csc^2 \theta}{(I^2+J^2-2)}\right)^N} d\theta \quad (14)$$

where E is the energy of the symbol and L represents I and J unified. When $k = 0$, $P_{b, \text{text Ray}}$ results in RQAM over Rayleigh channel.

3.3 Nakagami- m Fading Model

For Nakagami- m channel, the average symbol error probability has been derived in [5] as

$$P_{s, \text{Naka}} = \int_0^\infty f_R(r) P_S(r A_I, r A_Q) dr \quad (15)$$

$$P_{s, \text{Naka}} = 2K_I \psi(A_I) + 2K_Q \psi(A_Q) - 4K_I K_Q \Upsilon(A_I, A_Q) \quad (16)$$

where $K_I = (1 - 1/M_I)$ and $K_Q = (1 - 1/M_Q)$. Here M_Q and M_I are quadrature and in-phase signals. d_Q and d_I are the distances, σ_n^2 is the noise power, $A_Q = d_Q/\sigma_n$, and $A_I = d_I/\sigma_n$ with

$$\Psi(a) = \int_0^\infty Q(ar) f_R(r) dr \quad (17)$$

Closed form of $\psi(a)$ for integer value of m can be written as

$$\psi(a) = \frac{1}{2} - \frac{M(a)}{2} \sum_{k=0}^{m-1} \binom{2k}{k} \left(\frac{1 - M(a)^2}{4} \right)^k \quad (18)$$

with $M(\alpha) = \sqrt{\frac{z^2 \Omega}{1+z^2 \Omega}}$ and $\Omega = E\langle R^2 \rangle$, and

$$\begin{aligned} \Upsilon(a, b) &= \int_0^\infty \int_0^{\frac{b}{a}x_1} \int_0^{\frac{x_2}{b}} f(x_1) f(x_2) f_R(r) dr dx_2 dx_1 \\ &+ \int_0^\infty \int_0^{\frac{a}{b}x_2} \int_0^{\frac{x_1}{a}} f(x_2) f(x_1) f_R(r) dr dx_1 dx_2 \\ &= \Upsilon_1(a, b) + \Upsilon_1(b, a) \end{aligned} \quad (19)$$

where $f(z) = \frac{e^{-\frac{z^2}{2\pi}}}{\sqrt{2\pi}}$. From the bits $\log_2(I \cdot J)$, $\log_2(J)$ and $\log_2(I)$ are mapped onto quadrature channel and in-phase channel, respectively, and the amplitudes A_J and A_I are then chosen from the set $\{\pm d_j \pm 3d_j \dots \pm (J-1)d_j\}$ and $\{\pm d_i \pm 3d_i \dots \pm (I-1)d_i\}$, respectively. For $I \times J$ RQAM, error probability for k th and l th bit of the quadrature and in-phase, respectively, components is [4]

$$P_I(k) = \frac{1}{I} \sum_{i=0}^{(1-2^{-k})I-1} \left\{ w(i, k, I) \times \left(1 - \frac{\sqrt{\frac{3(2i+1)^2 \log_2(I \times J) \times \gamma}{I^2 + J^2 - 2}}}{\sqrt{\frac{3(2i+1)^2 \log_2(I \times J) \times \gamma}{I^2 + J^2 - 2} + 1}} \right) \right\} \quad (20)$$

$$P_J(l) = \frac{1}{J} \sum_{j=0}^{(1-2^{-l})J-1} \left\{ w(j, l, J) \times \left(1 - \frac{\sqrt{\frac{3(2j+1)^2 \log_2(I \times J) \times \gamma}{I^2 + J^2 - 2}}}{\sqrt{\frac{3(2j+1)^2 \log_2(I \times J) \times \gamma}{I^2 + J^2 - 2} + 1}} \right) \right\} \quad (21)$$

with

$$w(i, k, I) = (-1)^{\lfloor \frac{i \times 2^{k-1}}{I} \rfloor} \times \left(2^{k-1} - \left[\frac{i \times 2^{k-1}}{I} + \frac{1}{2} \right] \right) \quad (22)$$

$$w(j, l, J) = (-1)^{\lfloor \frac{j \times 2^{l-1}}{J} \rfloor} \times \left(2^{l-1} - \left[\frac{j \times 2^{l-1}}{J} + \frac{1}{2} \right] \right) \quad (23)$$

The equation between distance d_j and d_i , energy E_b in RQAM $I \times J$ system is

$$E_b = \frac{(I^2 - 1)d_I^2 + (J^2 - 1)d_J^2}{3 \log_2(I \times J)} \quad (24)$$

3.4 Log-Normal Fading Model

We have derived analytical expression of the ASEP for $N_I X N_Q$ RQAM schemes. Using PDF approach, AEP of the considered system may be computed by (3).

Here, $f_\gamma(\gamma)$ at the receiver can be represented as

$$f_\gamma(\gamma) = \frac{1}{\sqrt{2\pi}\sigma_{hT}\gamma} \exp\left(-\frac{\left(\ln\left(\frac{\gamma}{\gamma_0}\right) - 2\mu h_T\right)^2}{4\sigma_{hT}^2}\right) \quad (25)$$

The closed equation of ASEP for RQAM subject to log-normal fading is [6]

$$\begin{aligned} P_{s,LN} = & \frac{1}{\sqrt{2\pi}} \sum_{i=1}^n 2v_1 w_i Q\left(g1\sqrt{\gamma_0 \exp(2\sigma_{hT}x_i + 2\mu h_T)}\right) \\ & + \frac{1}{\sqrt{2\pi}} \sum_{i=1}^n 2v_2 w_i Q\left(g2\sqrt{\gamma_0 \exp(2\sigma_{hT}x_i + 2\mu h_T)}\right) \\ & - \frac{4v_1 v_2}{\sqrt{2\pi}} \sum_{i=1}^n 2v_1 w_i Q\left(q1\sqrt{\gamma_0 \exp(2\sigma_{hT}x_i + 2\mu h_T)}\right) \\ & \times Q(g2\sqrt{\gamma_0 \exp(2\sigma_{hT}x_i + 2\mu h_T)}) \end{aligned} \quad (26)$$

Here, x_i and w_i are zeroes and weights of the Hermite polynomial. h_T denotes channel estimation error, and σ^2 is the noise variance.

4 Simulation and Results

The performance of the modulation schemes and fading channels are simulated in Python. The M-ary square-QAM and rectangular QAM were simulated over the RF fading channels, and it was observed that indeed square-QAM performance was better than the rectangular QAM, though RQAM provides the robustness and flexibility as discussed above. Here, we simulated different orders of RQAM, viz. 4×2 , 8×2 , and 16×1 . The Rician channel as there is a Line of Sight between the transmitter and the receiver results in a lower BER as shown in Fig. 5 than the Rayleigh channel as shown in Fig. 4. The simulations for Nakagami-m channel are given in Figs. 7 and 8. The fact was observed that when the parameter m kept equal to 1, and it represents the case of Rayleigh fading. As the m parameter increases, it relates with the k parameter of Rician fading. When it further increases, it will converge to AWGN, i.e., no fading [7].

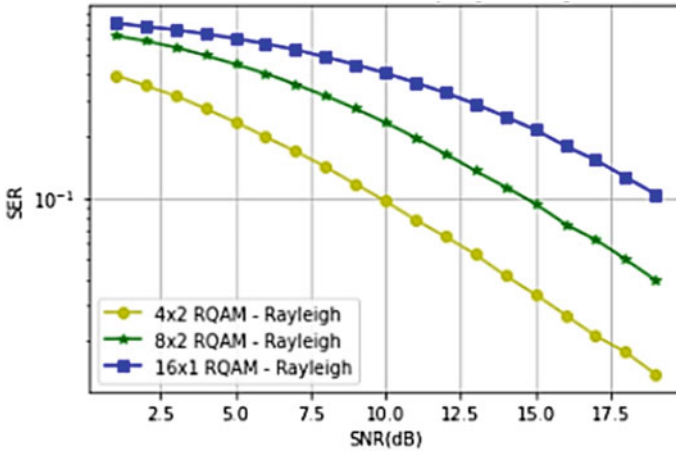


Fig. 4 Performance of RQAM over Rayleigh model

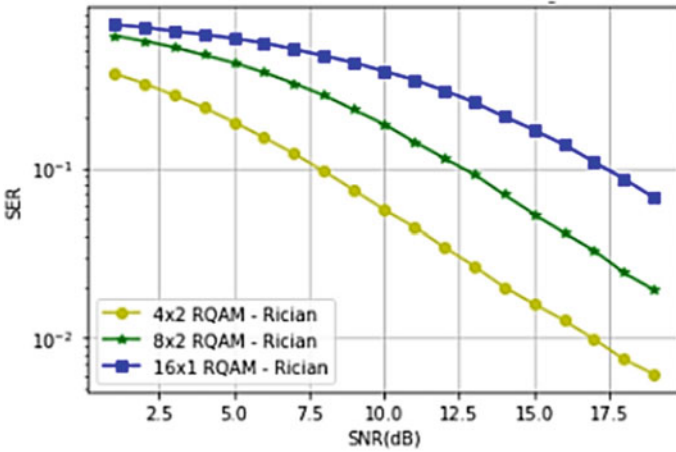


Fig. 5 Performance of RQAM over Rician model

As observed, the three schemes taken into consideration here showed a significant difference in the error rates at higher SNRs. For example, in Fig. 6, at 7.5 dB SNR, the error rates did not have huge difference for 16×1 and 8×2 schemes, but as going further, the error rates for 8×2 fell significantly than 16×1 scheme.

The error rates for 4×2 scheme remain significantly better than the others throughout. In Figs. 6 and 7, the error rates in Nakagami- m channel are shown, between when $m = 1$ and $m > 1$, the difference at lower SNRs remains significantly lower and increased uniformly as the SNRs increased, because of the fact that there

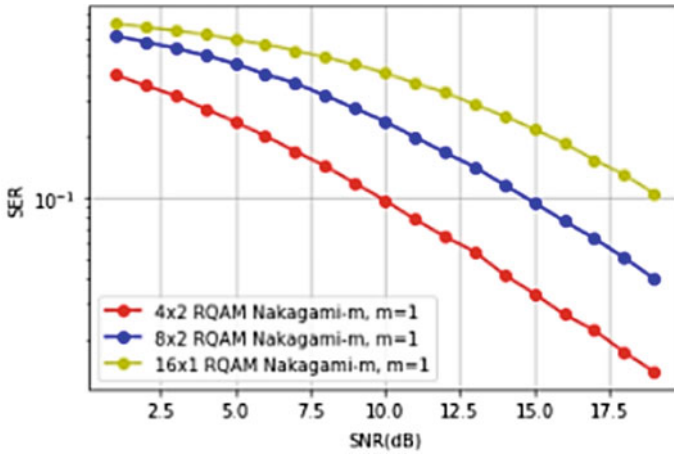


Fig. 6 Performance of RQAM over Nakagami-m model, where $m = 1$

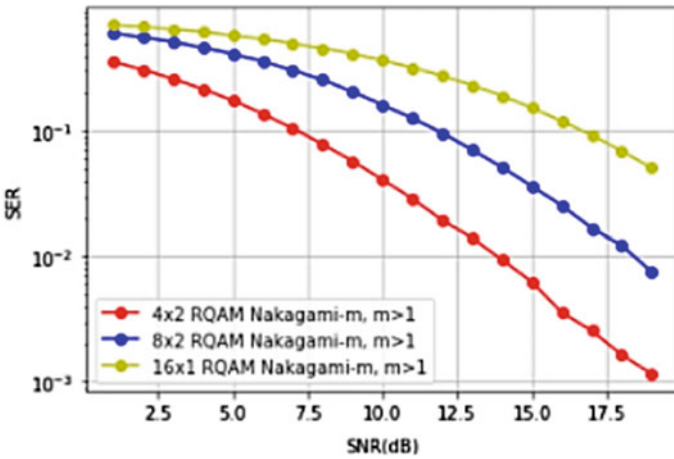


Fig. 7 Performance of RQAM over Nakagami-m model, where $m > 1$

is a dominant factor. In Fig. 8, the curves for 4×2 RQAM have been simulated for the log-normal fading model. In this case, a perfect CSI was assumed, and hence, at higher SNRs, the system gave a decent error rate. It was observed that after an SNR gain of 9 dB, there was a significant fall in the error rates [6].

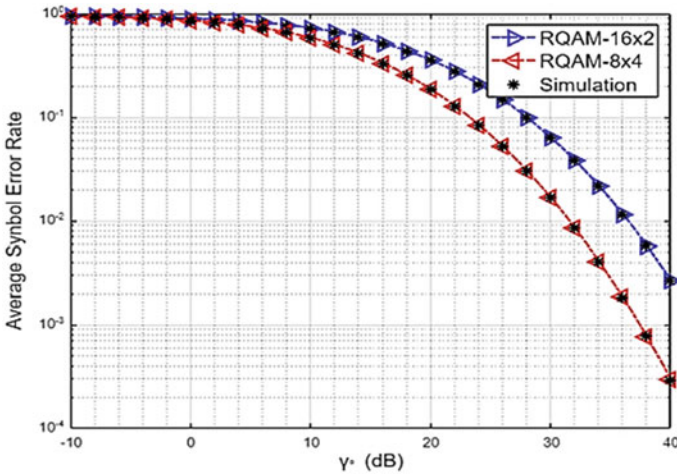


Fig. 8 Performance of RQAM over log-normal model

5 Conclusion and Future Work

The major part of this paper was to survey and understand the various RQAM techniques and its performance in various important RF and FSO channels. The RQAM modulation scheme gives a higher error rate than square-QAM, but has its own advantages in the aspects of security by segregating the bits and encoding them according to the priority. The RQAM scheme is much favored as it provides a greater flexibility and hosts a generic nature with inclusivity of other QAM and non-QAM schemes which can be implemented through it. For the fading channels, we observed Rayleigh, Rician, and Nakagami- m channel models in RF fading models, and we observed the weak turbulence representing log-normal fading channel in the optical fading models.

The observations derived from the theoretical analysis are verified using simulations in Python. This paper serves the ultimate purpose of concatenating the information of RQAM modulation scheme along with its variations and its performance analysis through various RF and optical fading models. This work can be further extended to the concept of adaptive modulation within the different levels and variations of RQAM, which may help in improving the reliability of communication links.

References

1. Simon MK, Alouini M (2008) Digital communications over fading channels, July 2008. <https://doi.org/10.1109/TIT.2008.924676>
2. Cho K, Yoon D, Jeong W, Kavehrad M (2001) BER analysis of arbitrary rectangular QAM. Conference on Record of Thirty-Fifth Asilomar conference on signals, systems and computers (Cat. No. 01CH37256)

3. Lopes WTA, Madeiro F, Alencar MS (2007) Closed-form expression for the bit error probability of rectangular QAM subject to Rayleigh fading. In: IEEE 66th Vehicular technology conference, pp. 915–919 (2007). <https://doi.org/10.1109/VETEFC.2007.200>
4. Sun J (2014) Linear diversity analysis for QAM in Rician fading channels. In: 23rd Wireless and optical communication conference (WOCC), vol. 2014, pp. 1–3 (2014). <https://doi.org/10.1109/WOCC.2014.6839960>
5. Karagiannidis GK (2006) On the symbol error probability of general order rectangular QAM in Nakagami-m fading. IEEE Commun Lett 10(11):745–747. <https://doi.org/10.1109/LCOMM.2006.060798> November
6. Sharma R, Trivedi YN (2021) Impact of imperfect CSI on the performance of inhomogeneous underwater VLC system. In: Proceedings of the International conference on paradigms of computing, communication and data sciences: PCCDS 2020, pp 287–298. Springer Singapore
7. Sharma R, Trivedi YN (2021) Performance analysis of dual-hop underwater visible light communication system with receiver diversity. Opt. Eng. 60(3):035111 (2021)

New Symmetric Key Cipher Based on Quasigroup



Umesh Kumar, Aayush Agarwal, and V. Ch. Venkaiah

Abstract Stream ciphers that use the XOR function for mixing the plaintext and the keystream are vulnerable to attacks such as known-plaintext attack and insertion attack. To overcome such shortcomings of the existing ciphers, we hereby propose a new stream cipher that uses AES. The proposed cipher is based on a large-order quasigroup. It is resistant to brute force attack, due to the exponential number of quasigroups of its order. It is also analyzed against the chosen-ciphertext, chosen-plaintext and known-plaintext attacks, and it is found to resist these attacks. The output of the cipher is subjected to various statistical tests, such as the NIST-STS test suite, and the results show a high degree of randomness of the ciphertext. Hence, it is resistant to correlation-type attacks.

Keywords AES · NIST-STS test · Latin squares · Quasigroup · Stream cipher

1 Introduction

The need for securing the data has been increasing day by day and with that there has been tremendous need for new encryption/decryption methods. A cryptosystem typically consists of an encryption algorithm, a decryption algorithm and a key generation algorithm. A cryptosystem may be analyzed to determine either the message or the secret key used.

Various cryptographic algorithms such as DES and AES are designed to achieve message confidentiality. Depending on how the data is encrypted, ciphers can be of two types: stream ciphers and block ciphers. Stream ciphers encrypt data bit by bit or byte by byte using a keystream which is as long as the plaintext and is generated

U. Kumar (✉) · V. Ch. Venkaiah
School of Computer & Information Sciences, University of Hyderabad, Hyderabad, India
e-mail: kumar.umesh285@gmail.com

V. Ch. Venkaiah
e-mail: vvcs@uohyd.ernet.in

A. Agarwal
Manipal Institute of Technology, Manipal, India

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
D. P. Agrawal et al. (eds.), *Cyber Security, Privacy and Networking*, Lecture Notes in Networks and Systems 370, https://doi.org/10.1007/978-981-16-8664-1_8

using a secret key. The keystream used for stream ciphers cannot be used again since stream ciphers are vulnerable to reused key attack [19, 20]. These ciphers are also vulnerable to attacks such as a known-plaintext attack and insertion attack since they use XOR function with plaintext and keystream [2]. The proposed algorithm is resistant to both these attacks since it uses quasigroup operation in place of XOR function and the keystream can be reused for encryption. Block ciphers encrypt a fixed size of data called blocks at one time. The size of the block depends on the encryption algorithm. DES and AES are examples of block ciphers. The DES was previously used as the standard for encryption. It was vulnerable to attacks such as brute force attack [4] because of its small key size of 56 bits, chosen-plaintext attack [3] and known-plaintext attack [11]. Hence, a new standard was required, and therefore, DES was replaced by AES [16].

AES is currently the standard for encryption and decryption. It is known to withstand various attacks. Though AES is highly secure, there is a need for an algorithm that is stronger than AES. Required promise comes from a mathematical object called a quasigroup.

Quasigroups are very simple non-associative algebraic structures. Since the number of quasigroups grows exponentially with the size, they make an important case for the design of cryptosystems. Using quasigroups, simple and efficient encryption algorithms can be produced. One of the factors that favor quasigroups is that they can be efficiently stored. Previous works [10, 15] that use quasigroups in the design of secure systems are vulnerable to the chosen-plaintext and chosen-ciphertext attacks [9, 21]. The proposed algorithm resists these attacks and hence is secure.

This paper proposes a new cipher algorithm for the encryption/decryption of the messages that replaces the XOR operation of the conventional stream ciphers by the quasigroup operation and its inverse. It is a stream cipher and uses AES for its keystream generation. In fact, any secure pseudorandom number generator such as CRT-DPR 4 [1] can be employed for the generation of the keystream. However, we choose to describe the proposed stream cipher using the AES 256. The security of the cipher is analyzed, and the randomness of the obtained ciphertext is tested. It was found that the proposed system satisfied all the required properties.

In this context, we would like to mention that similar scheme without any analysis was proposed in [5]. This scheme is based on deterministic finite automaton and uses Latin squares for encryption and decryption.

2 Preliminaries

In this section, a brief overview of quasigroups and their uses in the design of cryptosystems is discussed.

2.1 Latin Squares

A Latin square of order m is a $m \times m$ array in which the entries are taken from a finite set S and the symbols are arranged in such a way that each symbol occurs only once in each column and only once in each row.

Example 1: The following array (Table 1) is an example of a Latin square of order 4. It can be seen that each symbol occurs only once in each row and each column. The number of Latin squares of order n increases greatly as n increases. An estimate of the number of Latin squares of order n is given in [6, 7]

$$\prod_{k=1}^n (k!)^{\frac{n}{k}} \geq LS(n) \geq \frac{(n!)^{2n}}{n^{n^2}}, \tag{1}$$

where $LS(n)$ denotes the number of Latin squares of order n . For $n = 2^k, k = 7, 8$ the estimated number is:

$$0.164 \times 10^{21091} \geq LS(128) \geq 0.337 \times 10^{20666}, \tag{2}$$

$$0.753 \times 10^{102805} \geq LS(256) \geq 0.304 \times 10^{101724}. \tag{3}$$

2.2 Quasigroup

A quasigroup $Q = \langle S, * \rangle$ is a groupoid which has the following properties:

- (i) If a pair $a, b \in S$, then $a * b \in S$ (Closure property).
- (ii) $\forall a, b \in S$, there exists unique $x, y \in S$ such that $a * x = b$ and $y * a = b$.

Let $Q = \langle S, * \rangle$ be a quasigroup. Let \backslash (Left Inverse) and $/$ (Right Inverse) be two operations on Q such that

$$a * b = c \Leftrightarrow b = a \backslash c, \tag{4}$$

$$b * a = c \Leftrightarrow b = c / a, \tag{5}$$

Table 1 Latin square of order 4

0	1	2	3
1	2	3	0
2	3	0	1
3	0	1	2

Table 2 Operation table of Q

*	0	1	2
0	0	1	2
1	1	2	0
2	2	0	1

Table 3 Operation table of LIQ

\	0	1	2
0	0	1	2
1	2	0	1
2	1	2	0

Table 4 Operation table of RIQ

/	0	1	2
0	0	2	1
1	1	0	2
2	2	1	0

where a, b and c are elements of the quasigroup Q . Then, $LIQ = \langle S, \backslash \rangle$ and $RIQ = \langle S, / \rangle$ are called the Left Inverse and Right Inverse quasigroups of the quasigroup $Q = \langle S, * \rangle$, respectively.

Example 2: Consider the quasigroup $Q = \langle S, * \rangle$ with $S = \{0, 1, 2\}$, let its operation table be as in Table 2. Then the corresponding Left Inverse quasigroup is $LIQ = \langle S, \backslash \rangle$ whose operation table is given in Table 3. And the corresponding Right Inverse quasigroup is $RIQ = \langle S, / \rangle$ whose operation table is given in Table 4.

Properties (i) and (ii) of quasigroups enforce the operation table of a quasigroup to be a Latin square. Therefore, the number of quasigroups is same as that of the number of Latin squares. Therefore, Eq. 1 is also an estimate of the number of quasigroups. Hence, an estimate of the number of quasigroups of order 256 is given by Eq. 3.

2.3 Encryption and Decryption Using Quasigroups

Quasigroups are used for encryption purposes. Let $PT = p_1 p_2 p_3 \dots p_n$, $K' = k_1 k_2 k_3 \dots k_n$ and $CT = c_1 c_2 c_3 \dots c_n$ denote the plaintext to be encrypted, keystream to be used for encryption, and the resulting ciphertext, respectively. Then a way of encrypting PT with the keystream K' to obtain the corresponding CT is as follows: $c_1 = p_1 * k_1, c_2 = p_2 * k_2, \dots, c_n = p_n * k_n$.

For decryption of the ciphertext, the Right Inverse quasigroup $RIQ = \langle S, / \rangle$ is used. The following procedure decrypts the ciphertext CT , obtained from the foregoing encryption procedure, using the keystream K' to arrive at the corresponding plaintext PT . $p_1 = c_1/k_1, p_2 = c_2/k_2, \dots, p_n = c_n/k_n$.

Note that the conventional stream ciphers that exist in the literature use XOR operation in place of $*$ and $/$ operations. The algorithm works on characters of 1 byte (8 bits) each. Hence, all p_i, k_i, c_i are characters. Similarly, the message can be encrypted and decrypted with the quasigroup and its Left Inverse quasigroup, respectively.

Example 3: Consider the plaintext as $PT = p_1 p_2 p_3 = 021$ and the keystream to be $K' = k_1 k_2 k_3 = 011$. Then applying the foregoing encryption procedure using the quasigroup Q (Table 2) of example 2, we have the ciphertext as

$$CT = c_1 c_2 c_3 = 002.$$

To decrypt the ciphertext, Right Inverse Quasigroup RIQ (Table 4) is used. The recovered plaintext will be

$$PT = p_1 p_2 p_3 = 021.$$

Other methods of encryption/decryption using quasigroups are addressed in [7].

2.4 Advanced Encryption Standard

Advanced Encryption Standard or AES is a symmetric key block cipher. It has 128-bit data with 128/192/256-bit key. AES is widely used for encryption/decryption process in various fields. It has fast implementation in both software and hardware. AES is used in various modes of operation such as Cipher Block Chaining (CBC) mode, Cipher Feedback (CFB) mode, Output Feedback (OFB) and Counter (CTR) mode. Each mode of operation has their advantages and disadvantages [8, 18].

3 Proposed Cipher Algorithm Structure

3.1 Quasigroup Selection

The main principle used is the quasigroup operation of 1-byte plaintext characters with 1-byte random keystream characters to produce 1-byte ciphertext characters at a time. Any order quasigroup can be used in the algorithm. Since as the order increases the number of quasigroups grows exponentially, higher-order quasigroups are preferred. The order of 256 is used in our proposed algorithm because all the

ASCII values can be represented in 8 bits and each character has an integer value of 0 to 255. Also, from Eq. 3 we can see that the number of quasigroups of order 256 is very large and therefore practically impossible to guess the correct quasigroup selected. That is, it is impossible to determine the quasigroup $Q = \langle S, * \rangle$, where $S = \{0, 1, 2, \dots, 255\}$ and $*$ is the binary operation used in our algorithm. The issue of generating large-order quasigroups is addressed in [13, 14].

3.2 Keystream Generation

The encryption/decryption algorithm uses a keystream of size as long as the message. To generate such a long keystream, the algorithm uses AES. Using an initialization vector (IV), a secret key (K), and a *Counter*, the AES algorithm produces keystream of required length. The keystream generated is represented by K' . Since the encryption algorithm requires the keystream to be random, AES ensures this. Block diagram of the keystream generation is given in Fig. 1.

Note that the secret key K used in AES is different from the keystream K' , generated using AES. Each round generates 16 bytes of keystream and is repeated until the keystream size is the same as that of the plaintext. This generates the keystream $K' = k_1 k_2 k_3 \dots k_n$, where each k_i is a 1-byte character and will be used to encrypt 1-byte character of the plaintext. The keystream generation can use AES in encryption or

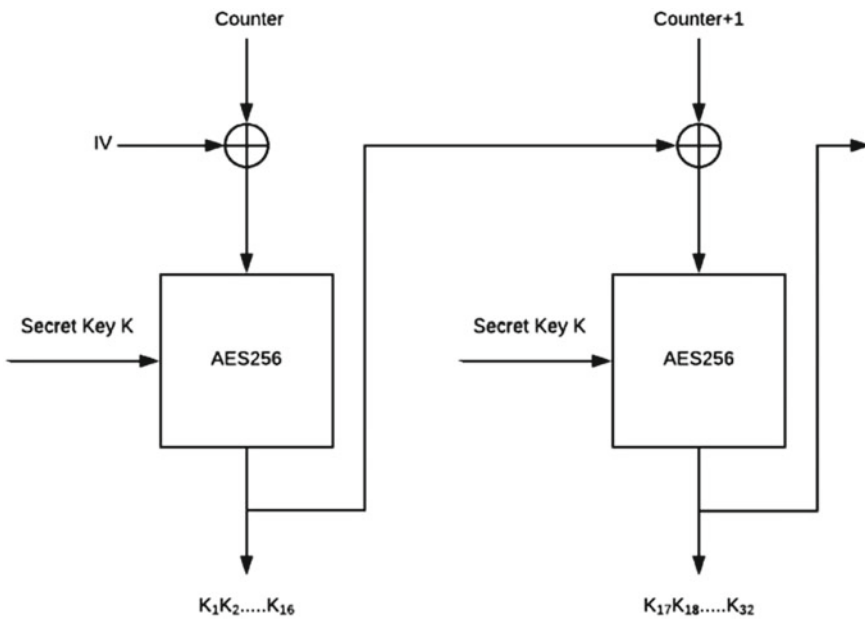
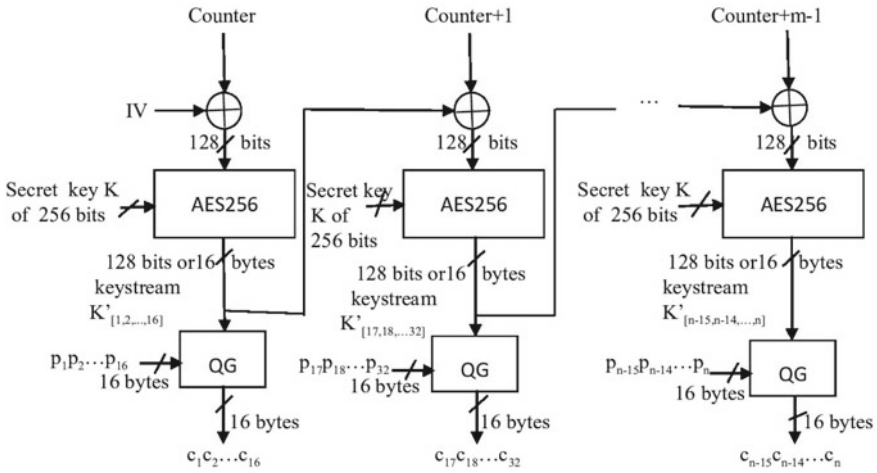


Fig. 1 Keystream generation using AES 256



Note that m is the number of plaintext blocks. Size of each block is 16 bytes.

Fig. 2 Encryption algorithm

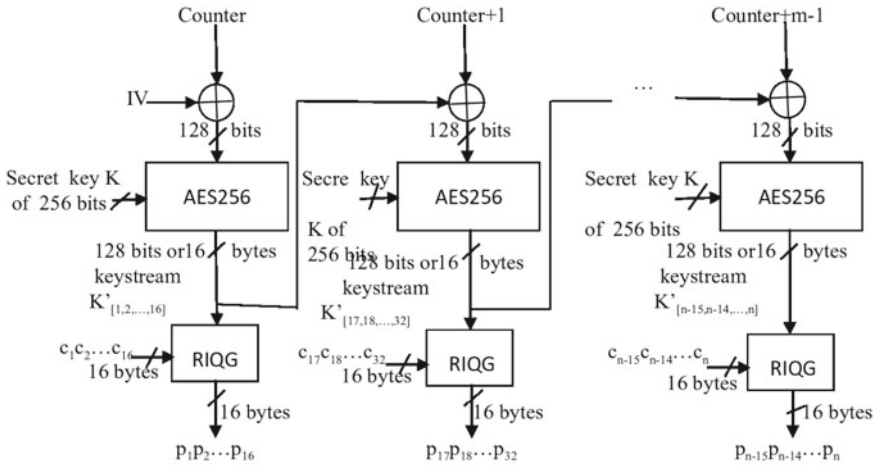
decryption mode. The keystream generation procedure is used in both encryption and decryption processes of the proposed stream cipher. The secret key size can be 128/192/256 bits. The AES algorithm used in the proposed cipher is AES 256 encryption algorithm.

3.3 Encryption Algorithm

The encryption algorithm uses the method described earlier. The main principle used is as follows: For plaintext, $PT = a_1 a_2 a_3 \dots a_n$ and the generated keystream, $K' = k_1 k_2 k_3 \dots k_n$ the ciphertext, $CT = c_1 c_2 c_3 \dots c_n$ is produced as $c_1 = a_1 * k_1, c_2 = a_2 * k_2, \dots, c_n = a_n * k_n$, where '*' is the operation of the chosen quasigroup. The algorithm can encrypt 16 bytes of plaintext in one iteration. Block diagram of the encryption algorithm is given in Fig. 2.

3.4 Decryption Algorithm

The decryption algorithm uses the inverse of the quasigroup $Q = \langle S, * \rangle$ chosen for encryption. Depending on the operation used in encryption either the Left Inverse or the Right Inverse can be used to generate the plaintext PT . The main principle used is as follows: For ciphertext, $CT = c_1 c_2 c_3 \dots c_n$ and the keystream, $K' = k_1 k_2 k_3 \dots k_n$ the corresponding plaintext $PT = a_1 a_2 a_3 \dots a_n$ can be recov-



Note that m is the number of ciphertext blocks. Size of each block is 16 bytes.

Fig. 3 Decryption algorithm

ered as $a_1 = c_1/k_1, a_2 = c_2/k_2, \dots, a_n = c_n/k_n$, where $'/'$ is the Right Inverse of the operation of the chosen quasigroup. The algorithm uses the same secret key as that of the encryption algorithm. It can decrypt 16 bytes of ciphertext in one iteration. The cipher algorithm uses the Right Inverse quasigroup $RIQ = \langle S, / \rangle$ for decryption. Block diagram of the decryption algorithm is given in Fig. 3.

4 Security Analysis

The key elements for the algorithm consist of the secret key K , the initialization vector IV , the *Counter*, and the selected quasigroup Q of order 256. Since the secret key size is 256 bits, it is resistant to brute force approach since there are 2^{256} possible keys. Since the keystream generation for the proposed stream cipher is similar to AES in Cipher Block Chaining (CBC) mode encryption, it is already secure from attacks. Hence, the keystream K' generation is safe from various attacks.

The other element is the quasigroup. Since the order of used quasigroups is 256, the number of possible quasigroups is at least

$$0.304 \times 10^{101724}.$$

Therefore, it is impossible to determine the selected quasigroup. The general method given in [10, 15] used for encryption using quasigroups is found to be vulnerable to chosen-ciphertext attack and chosen-plaintext attack [9, 21], whereas the proposed algorithm is resistant to these attacks because of the following argument: Suppose the

cryptanalyst chooses the ciphertext $CT = c_1 c_2 c_3 \dots c_n$, and obtains the plaintext $PT = p_1 p_2 p_3 \dots p_n$, corresponding to the chosen-ciphertext and tries to determine the quasigroup Q employed in the encryption-decryption process. The adversary, then, for the keystream K' , must solve the system of equations:

$$\begin{aligned} c_1 &= p_1 * k_1, \\ c_2 &= p_2 * k_2, \\ &\dots \\ c_n &= p_n * k_n, \end{aligned}$$

where

$$K' = k_1 k_2 k_3 \dots k_n$$

is the generated keystream. This system has as many solutions as there are quasigroups of order 256. Hence, determining the quasigroup makes it practically impossible. Therefore, the cipher is resistant to chosen-ciphertext attack. Even if the attacker has knowledge of both the plaintext and the corresponding ciphertext, the cipher still remains secure as determining K' still requires finding the selected quasigroup. Therefore, it is resistant to known-plaintext attack as well. Similar argument shows that the system is secure from the chosen-plaintext attack also. The decryption algorithm uses the Right Inverse (same can be achieved using Left Inverse LIQ quasigroups as well) of the quasigroup Q denoted by RIQ . The cipher encrypts/decrypts a stream of characters; hence no padding of plaintext is required and hence is safe from padding oracle attacks which are a known threat to ciphers where padding is required as shown in [12].

Note that the computational complexity is exactly the same as any stream cipher that uses AES for keystream generation and XOR function for mixing the plaintext and the keystream. Except that it differs in the following: Existing ciphers use XOR function for every bit of the message; whereas our cipher works on a character by character and uses one quasigroup operation for every character of the message. The space complexity of our cipher is also the same as that of the existing ciphers, except that our cipher needs one quasigroup table of 256×256 . That is, our cipher needs 64k bytes of extra space.

4.1 Statistical Test for Randomness

The obtained ciphertext after encryption with the proposed algorithm passes various tests of randomness. One such battery of tests is the NIST-STS test suite [17]. Each test of the NIST-STS package gives a P-value which lies between 0 and 1 (both included) and indicates Success/Fail status. The P-value is the probability that a perfect random number generator would have produced a less random sequence than the one being tested [17]. For these tests, we have chosen the significance level (α)

Table 5 Parameters for the NIST-STS test

Tests	Block length(m)
Block frequency test	128
Non-overlapping template test	9
Overlapping template test	9
Approximate entropy test	10
Serial test	16
Linear complexity test	500

Table 6 Results of the NIST-STS test

Tests	<i>P</i> -value
Frequency	0.507678
Block frequency	0.513950
Cumulative sums-forward	0.675720
Cumulative sums-backward	0.530005
Runs	0.542138
Longest run	0.552834
Rank	0.443481
Discrete Fourier transform	0.500707
Overlapping template	0.479753
Approximate entropy	0.471052
Serial-1	0.550659
Serial-2	0.553913
Linear complexity	0.576939

to be 0.01 and the other parameters as shown in Table 5. For the randomness of a sequence, we compare the *P*-value of a sequence to a significance level (α). If *P*-value $\geq \alpha$, then the sequence is considered to be random, otherwise non-random. We run each of these tests over 20 obtained ciphertext sequences. The size of each sequence is 200 KB. Table 6 shows the average *P*-value of NIST-STS tests. We can observe that the *P*-value of each test crosses the significance level ($\alpha = 0.01$), so, we conclude that the obtained ciphertext sequences are random.

5 Conclusion

A new stream cipher algorithm that uses the existing cipher for keystream generation and a quasigroup for encryption and decryption is proposed. It masks the weaknesses of stream ciphers and adds extra security. The new cipher is resistant to most common

attacks such as chosen-ciphertext attack, chosen-plaintext attack and known-plaintext attack. The randomness of the obtained ciphertext is analyzed by the NIST-STS test suit, and we noted that the new cipher produces high degree of randomness of the ciphertext. The keystream in our proposed cipher as in the case of any stream cipher can be preprocessed and kept ready, and then the encryption/decryption can be performed parallelly. Storing of the generated key can be a challenge for very large messages since the generated key is as long as the message. Our cipher can be deployed in all the applications of stream ciphers such as secure wireless connection.

References

1. Barker E, Kelsey J (2007) Recommendation for random number generation using deterministic random bit generators. Technical report, NIST (revised)
2. Bayer R, Metzger J (1976) On the encipherment of search trees and random access files. *ACM Trans Database Syst (TODS)* 1:37–52
3. Biham E, Shamir A (1993) *Differential cryptanalysis of the data encryption standard*. Springer, Berlin
4. Diffie W, Hellman ME (1977) Special feature exhaustive cryptanalysis of the NBS data encryption standard. *Computer* 10:74–84
5. Domosi H (2017) A novel stream cipher based on deterministic finite automaton. Ninth workshop on non-classical models of automata and applications (NCMA 2017), pp 11–16
6. Jacobson MT, Matthews P (1996) Generating uniformly distributed random Latin squares. *J Combin Des* 4:405–437
7. Koscielny C (2002) Generating quasigroups for cryptographic applications. *Int J Appl Math Comput Sci* 12:559–570
8. Lipmaa H, Wagner D, Rogaway P (2000) Comments to NIST concerning AES modes of operation: CTR-mode encryption
9. Maljutina NN (2019) Cryptanalysis of some stream ciphers. *Quasigroups Rel Syst* 27:281–292
10. Markovski S, Gligoroski D, Andova S (1997) Using quasigroups for one-one secure encoding. In: *Proceedings of VIII Conference on logic and computer science “LIRA”*, vol 97, pp 157–162
11. Matsui M (1993) Linear cryptanalysis method for DES cipher. In: *Workshop on the theory and application of cryptographic techniques*. Springer, Berlin, pp 386–397
12. Paterson KG, Yau A (2004) Padding oracle attacks on the ISO CBC mode encryption standard. In: *Cryptographers’ track at the RSA conference*. Springer, Berlin, pp. 305–323
13. Petrescu A (2007) Applications of quasigroups in cryptography. In: *Proceedings of interdisciplinarity in engineering*. TG-Mures, Romania. Academic Press
14. Petrescu A (2009) A 3-quasigroup stream cipher. In: *The international conference interdisciplinarity in engineering INTER-ENG*. Editura Universitatii “Petru Maior” din Tirgu Mures, p 168
15. Petrescu A (2010) n-quasigroup cryptographic primitives: stream ciphers. *Stud Univ Babeş-Bolyai Inf* 55 [On table of contents: *Anul LIV*]:27–34
16. Rijmen V, Daemen J (2001) Advanced encryption standard. In: *Proceedings of federal information processing standards publications*. National Institute of Standards and Technology (NIST), pp 19–22
17. Rukhin A, Soto J, Nechvatal J, Smid M, Barker E (2010) A statistical test suite for random and pseudorandom number generators for cryptographic applications. NIST, special publication 800-22, revision 1a
18. Schneier B (2007) *Applied cryptography protocols, algorithms, and source code in C*. Wiley
19. Stallings W (2006) *Cryptography and network security*, 4th edn. Pearson education, Inc., India

20. Stinson D (1995) *Cryptography: theory and practice*. CRC Press, CRC Press LLC
21. Vojvoda M (2004) *Stream ciphers and hash functions-analysis of some new design approaches*. Ph.D. thesis. Slovak University of Technology

Validate Merchant Server for Secure Payment Using Key Distribution



A. Saranya and R. Naresh

Abstract Nowadays, growing and competitiveness of a financial sector and e-healthcare application developed rapidly and quicker ever so that mobile payment plays a vital role to make easier and quicker. In the existing paper, the cloud transmits the hash value straight to the merchant server without checking for fraud. This paper aims to provide a secure authentication mechanism between the cloud and the merchant server to avoid fraudulent merchant servers. For trustworthiness, our proposed system introduces secure key distribution between cloud and merchant server named as validate merchant server key distribution protocol (VMSKDP). Our proposed system improves security for mobile payments and reduces attacks like man-in-middle attacks, denial-of-service, etc.

Keywords Mobile payments · Merchant server · Key distribution · Security · Secure key

1 Introduction

All throughout the world, Smartphone mobiles have rapidly increased mobile phone applications and have provided better value-associated services for intelligent mobile phone customers and organizations. Hence these versatile devices the world at large performs day-to-day activities such as transferring payments, data sharing, and more. While introducing mobile phones to users, the assurance for proper functioning and efficiency of the applications needs to be stated as well. In addition, this should provide an additional guarantee that operating system techniques customize on smartphone customers, preferably digital phones users [1–3]. For mobile OS applications,

A. Saranya · R. Naresh (✉)
Department of Computer Science and Engineering, SRM Institute of Science and Technology,
Kattankulathur, Chengalpattu, Chennai, Tamil Nadu 603203, India
e-mail: nareshr@srmist.edu.in

A. Saranya
e-mail: sa1096@srmist.edu.in

Android OS is among the better prevalent stages and that has positively inspired a group of various mobile applications in the entire world. For the systematic practice of Android-based phone apps, Android-based smartphone vendors and cloud service providers [4, 5] must tackle novel tasks related to security and source administration. All information concerning mobile payments [6, 7] passes through a cloud environment. So, there is a need to focus on the module to provide better security and privacy [8–12]. There are so many challenges to secure transfer payments through mobile phones; the first challenge is “security,” the second one is “privacy,” and the third one is “availability.” Nowadays, all technologies may be a chance to provide all three challenges. Since the key distribution between a mobile user and merchant server is very problematic in the new world. So, our proposed system model offers more security in the key distribution phase and protects the transaction between the mobile user and merchant server. The subsequent sections are the outline of this paper; Sect. 2: Related works, Sect. 3: Execution of the System model secure key transmission and mobile payment, Sect. 4 stated the proposed model’s security analysis; Sect. 5 outlines the conceptual model’s performance analysis; Sect. 6 conclusions and future works.

1.1 The Objectives of the Proposed Work Are

- The key distribution between admin server, merchant server, and cloud environment.
- Cloud environment matching payment between mobile user and merchant server.
- Mobile user payment request encryption.

2 Related Works

Taparia et al. [13] proposed “Secure Key Exchange Using Enhanced Diffie–Hellman Protocol Based on String Comparison,” combining commitment and authentication scheme management to improve security on wired and wireless between two host computers and protect from man-in-the-middle threats.

Tasi et al. [14] developed a concept named “Weaknesses and improvements of an efficient certificateless signature scheme without using bilinear pairings,” providing a unique signature scheme to handle the problem of sharing keys using an identity-based signature scheme. Wang et al. [15] outlined a design named “A Modified Efficient Certificateless Signature Scheme without Bilinear Pairings,” providing a new scheme to achieve Yeh et al.’s model security level, including cracking all existing methods only.

Gong and Li [16] developed a concept named “Further improvement of a certificateless signature scheme without pairing,” providing more security against super adversaries under random oracle technology. He et al. [17] made a proposal named

“An efficient and provably secure certificateless signature scheme without bilinear pairings.” The goal is to minimize the system’s computational costs. Tsai et al. [18] outlined a design named “A New Efficient Certificateless Short Signature Scheme Using Bilinear Pairings,” aiming to reduce the verification process, computation cost and signature generation.

Bodur et al. [19] developed a concept named “Implementing Diffie–Hellman Key Exchange Method on Logical Key Hierarchy for Secure Broadcast Transmission” to perform key exchange using virtual key hierarchy of a broadcast communication channel.

In “Hybrid Context-Aware Recommendation System for E-Health Care by Merkle Hash Tree from Cloud using Evolutionary Algorithm”, Deepa et al. [20] describe a design that uses the Merkle tree feature to store patient data records outside of a cloud to minimize search time while maintaining high-security levels. Sung et al. [21] made a proposal named “User Authentication Using Mobile Phones for Mobile Payment,” aiming to provide a software token to the Transaction Certificate Mode (TCM) protocol. Mutual authentication should support and consider robbed, borrowed and corrupted smartphones for digital money.

Saranya and Naresh [22] outlined a design named “Cloud-Based Efficient Authentication for Mobile Payments using Key Distribution Method.” The cost of computation is kept in the payment section which improves efficiency. Because the Merchant Server desires to compute for each transaction process in the payment area, it enforced the assumption of batch verification to ease the problems that arise when many customers use the transaction area, allowing the Merchant Server to resolve the scalability issue.

Saranya and Naresh [23] developed a concept named “Efficient mobile security for E-healthcare application in cloud for secure payment using key distribution,” a mobile payment using the Secure Authentication Protocol (SAP) using cryptographic approaches to achieve shared authentication between the server and the client, which can be used to attack forged servers and false workstations. The security of user data and distinct privacy during the payment is to concurrently achieve security and maintain the practice acceptability of mobile payments within untrustworthy public communication systems, which is a vital matter for smart mobile device producers as well as mobile data consumers [24–27].

3 System Model

Our proposed system model discussed in this chapter is illustrated with components present. There are four components: admin server, merchant server, mobile user, and cloud environment as per Fig. 1.

As per Fig. 1, system model flow like (1) merchant server sends a registration request to admin server, (2) admin server sends secure hash value to respective register merchant server, (3) admin server sends the encrypted value to cloud using a secure channel, (4) merchant server sends hash value cloud, (5) mobile user sends

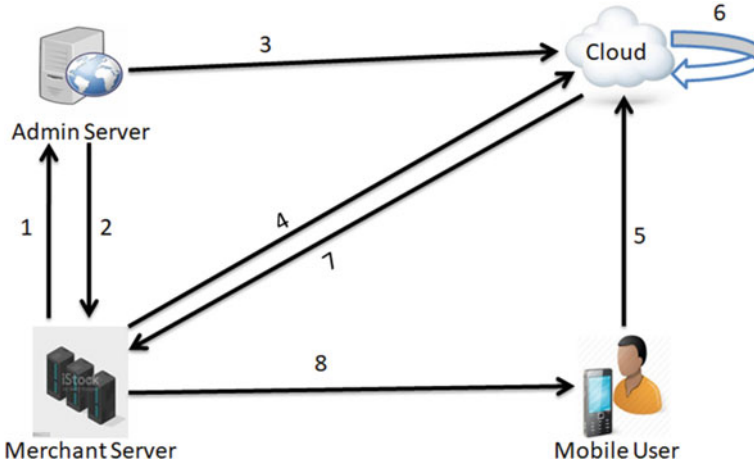


Fig. 1 Proposed system model

payment request using trapdoor generation function, (6) cloud matches hash value both of mobile user and merchant server, (7) cloud sends a payment request to verified respective merchant server and (8) merchant server sends the payment status, i.e., success/fail (Fig. 1).

3.1 Bilinear Mapping

Let G have a monogenous group with “ a ” in the prime order. “ c ” refers to the cyclic group function, and “ e ” refers to the bilinear pairing function $e = G \times G = G_1$. It should satisfy the three conditions for the bilinear pairing function, such as

$$(1) \text{ Bilinear } e(g^m, g^n) = e(g, g)^{mn}, \forall m, n \in \mathbb{Z}_p^*$$

$$(2) \text{ Non-degeneracy } g \in G \text{ i.e. } e(g, g) \in G$$

$$(3) \text{ “}e\text{” can be computed easily.}$$

3.2 Merchant Server Registration Process

In this section, the registration process of the merchant server using the Diffie–Hellman cryptosystem is discussed. Diffie–Hellman uses a multiplicative group of integers modulo q , where q is the exceptional values and g is the primitive root

modulo q . Based on the two values, our resulting mutual secret will take any value from 1 to $q - 1$. Initially, the merchant server and admin server agree the below formula to share the secret key for encryption and decryption of their data while registration.

$$(g^m \bmod q)^n \bmod q = (g^n \bmod q)^m \bmod q \quad (1)$$

After the verification process is successful, the merchant server uses “bankid” with a secret key to perform XOR operation to generate cipher text. The merchant server cipher text formula is as follows,

$$\text{Merchant Server Registration (MSR)} = \text{bankid} \oplus \text{shared secret key}$$

Each request generates a new shared private key, which makes replay attacks impossible on the merchant’s end.

3.3 Admin Server Process

In this section, the admin server process is discussed; with the help of the merchant server request, the admin decrypts the cipher text to the original value like below.

$$\text{Admin Server Decryption (ASD)} = \text{Ciphertext} \oplus \text{shared secret key} \quad (2)$$

After decryption, the cipher text admin sends a secure hash function to the merchant server and cloud environment using a secure channel.

$$\text{Admin server cloud hash value (ASHV)} = (g^{\text{bankid}}) \quad (3)$$

$$\text{Merchant server hash value (MSHV)} = (g^{H(\text{bankid}_{id})} \cdot g)^{\text{bankid}} \quad (4)$$

The above equation needs to be shared with the merchant server using a secure channel. After receiving the hash value, the merchant server needs to share the cloud environment for storing the proposal. Because based on user request, it matches correct merchant details.

3.4 Payment Request from Mobile User

In this section, the payment request is discussed. While the user is triggering the payment request, requested data should have to be processed using the trapdoor generation method. As we transfer plain texts, there are chances of the requests

being traced and consequently becoming open to attacks. So, for avoiding the attack, the trapdoor generation process is used to improve security. Generating payment requests is formulated as

$$\text{Patient Payment Request (PPR)} = (g^{H(\text{bankid}_{id}).\text{prn}}, g^{\text{prn}}) \quad (5)$$

where,

$$\text{Pr 1} = g^{H(\text{bankid}_{id}).\text{prn}} \text{ and } \text{Pr 2} = g^{\text{prn}}$$

3.5 Cloud Matching Process

In this section, we discussed the details about the cloud matching process. After receiving the entire request to the cloud, now the cloud needs to match and verify the merchant details and mobile user requests like below.

$$\text{Cloud payment request matching} = e(\text{PPR}, \text{MSHV}) \quad (6)$$

$$= e\left(\left(g^{H(\text{bankid}_{id}).\text{prn}}\right)^{\text{bankid}}, g^{\text{prn}}\right) \quad (7)$$

$$= e(g, g)^{(H(\text{bankid}_{id})+1).\text{bankid}.\text{prn}} \quad (8)$$

$$= e(g, g)^{H(\text{bankid}_{id}).\text{bankid}.\text{prn}} . e(g, g)^{\text{bankid}.\text{prn}} \quad (9)$$

$$= e\left(g^{H(\text{bankid}_{id}).\text{prn}}, g^{\text{bankid}}\right) . e\left(g^{\text{bankid}}, g^{\text{prn}}\right) \quad (10)$$

$$e(\text{PPR}, \text{MSHV}) = e(\text{Pr 1}, \text{ASHV}) . e(\text{ASHV}, \text{Pr 2}) \quad (11)$$

If (3.5) equation “true,” then only the cloud sends the payment request to merchant server using secure channel else cloud rejects the payment request, and once again user needs to generate an additional request (Table 1).

4 Security Analysis of System Model

The proposed system’s security measures have been listed in this chapter.

4.1 Man-in-Middle Attack

Whenever the user executes the man-in-the-middle assault in between the smart-phone network and the cloud, data communication between mobile user and cloud environment creates a trapdoor generation feature to perform two modules:

$$Pr 1 = g^{H(\text{bankid}_{id}).prn} \text{ and } Pr 2 = g^{prn}$$

As a result, while he wants to attack the data, he attempts to handle the DL algorithm and hash function

4.2 Impersonation Attack

An impersonation intrusion seems workable for an intruder who demands a payment number and then alters the payment request number while submitted by a legitimate mobile user. The intruder then grabs search queries similar to an authorized patient like

$$e(PPR, MSHV) = (g^{H(\text{bankid}_{id}).prn}, g^{\text{bankid}}).e(g^{\text{bankid}}, g^{prn})$$

where bankid_{id} . The payment request is bank number, and PRN is the random payment request number for the mobile user. Because of an NP-hard problem, an impersonation attack cannot estimate for our proposed system.

5 Performance Analysis

This chapter discusses the proposed system performance analysis up to a 128-bit sizeable prime number. Windows 10 OS and python are used for key generation,

Table 1 Notation for the proposed system model

S. No.	Notation	Description
1	<i>MSR</i>	<i>Merchant server registration</i>
2	<i>Bankid</i>	Bank ID based on the registered bank
3	<i>Shared secret key</i>	<i>shared secret key used to encrypt and decrypt</i>
4	<i>ASD</i>	<i>Admin server decryption</i>
5	<i>Ciphertext</i>	Encrypted value
6	<i>H (bankid_{id})</i>	Hash value with bank id
7	<i>prn</i>	Payment request number
8	<i>Pr1, Pr2</i>	Trapdoor request value

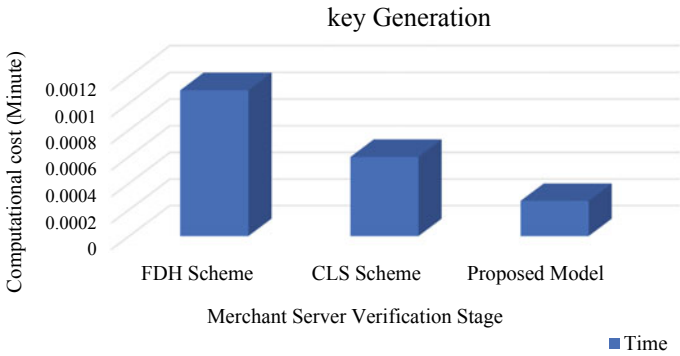


Fig. 2 Key generation proposed model

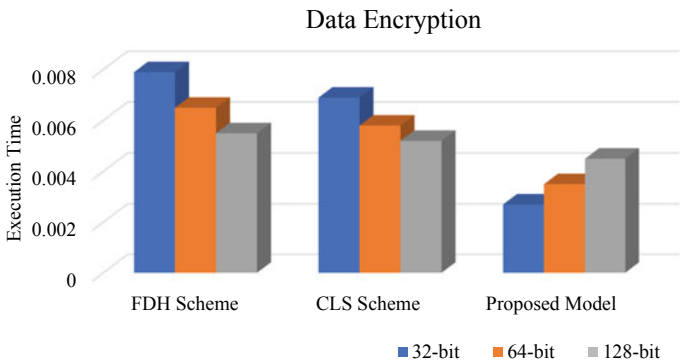


Fig. 3 Data encryption between admin and merchant server proposed model

encryption, and decryption as well that takes place between admin and merchant server that is based on the generated key. The encryption and decryption are done between the cloud and the merchant’s requests using the payment number and the bankid. All encryption and decryption are performed using the XOR operation and the transfer of encrypted data is done by using a secure channel between the admin and the server.

Below figure is key generation, encryption and decryption of our proposed model.

As per Fig. 2, this performs the key generation phase between admin and merchant server. Because we established a secure connection between the admin server and the cloud, the merchant server exchanges encrypted messages with the admin server to verify the merchant’s authorization.

As per the above figure, we computed up to 128-bit large prime numbers to generate shared secrets between admin and merchant server.

As per Fig. 3, data encrypted between admin and merchant server using a shared secret key is generated earlier. So, encrypted data is shared using a secure channel

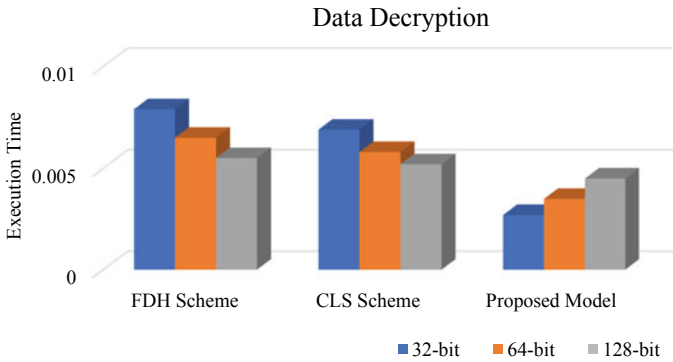


Fig. 4 Data decryption between admin and merchant server proposed model

between admin and merchant server and refers to the image time increase based on the shared private key.

As per Figs. 3 and 4, the encryption time and decryption time are same because of using XOR operation for both admin and merchant server.

6 Conclusions and Future Works

In this chapter, observations and future enhancement of a current proposal system are discussed. Our suggested architecture provides two ways for merchant servers and mobile users to authenticate. First, the merchant authenticates registration data on the admin server, and then the mobile user requests payment. If the merchant is available in the cloud, it sends the payment request to the merchant server to process. Even intruders cannot find the user request because of trapdoor generation for the user request. In our future enhancement of the proposed model, we proposed the key generation up to the 128-bit process. So, our system tried to improve the key generation level. And also, it tried to improve security level between admin and merchant server.

References

1. Huang X, Susilo W, Mu Y, Zhang F (2005) On the security of certificateless signature schemes from Asiacypt 2003. In: Proceedings of Fourth International Conference on cryptology and network security (CANS '05), pp 13–25
2. Xiong H (2014) Cost-effective scalable and anonymous certificateless remote authentication protocol. *IEEE Trans Inf For Secur* 9(12):2327–2339
3. Coron JS (2000) On the exact security of full domain hash. *Advances in cryptology-CRYPTO 2000. LNCS 1880*, pp 229–235

4. The Pairing-Based Cryptography Library (PBC). <https://crypto.stanford.edu/pbc/>
5. Katz EJ (2008) Hand book of mobile communication studies. The MIT Press
6. Alipay. <https://www.alipay.com/>
7. WeChatpay. <https://pay.weixin.qq.com/index.php/public/wechatpay>
8. Xiong H, Mei Q, Zhao Y (2018) Efficient and provably secure certificateless parallel key-insulated signature without pairing for IIoT environments. *IEEE Syst J*. <https://doi.org/10.1109/JSYST.2018.2890126>
9. Yeh KH (2017) A secure transaction scheme with certificateless cryptographic primitives for IoT-based mobile payments. *IEEE Syst J*. <https://doi.org/10.1109/JSYST.2017.2668389>
10. Xiong H, Zhang H, Sun J (2018) Attribute-based privacy-preserving data sharing for dynamic groups in cloud computing. *IEEE Syst J*. <https://doi.org/10.1109/JSYST.2018.2865221>
11. Qin Z, Sun J, Wahaballa A (2017) A secure and privacy-preserving mobile wallet with out-sourced verification in cloud computing. *Comput Stand Interfaces* 54:55–60
12. Bellare M, Garay J, Rabin T (1998) Batch verification of short signatures. In: International conference on the theory and applications of cryptographic techniques, pp 236–250
13. Taparia A, Panigrahy SK, Jena SK (2017) Secure key exchange using enhanced Diffie Hellman protocol based on string comparison. *IEEE WiSPNET 2017 conference*
14. Tasi J-L, Lo N-W, Wu T-C (2012) Weaknesses and improvements of an efficient certificateless signature scheme without using bilinear pairings. *Int J Commun Syst*
15. Wang L, Chen K, Long Y, Mao X, Wang H (2015) A modified efficient certificateless signature scheme without bilinear pairings. In: 2015 International conference on intelligent networking and collaborative systems
16. Peng G, Ping L (2014) Further improvement of a certificateless signature scheme without pairing. *Int J Commun Syst* 27:2083–2091
17. He D, Chen J, Zhang R (2011) An efficient and provably-secure certificateless signature scheme without bilinear pairings. *Int J Commun Syst*
18. Tsai J-L, A new efficient certificateless short signature scheme using bilinear pairings. *IEEE Syst J*
19. Bodur H, Kara R (2017) Implementing Diffie-Hellman key exchange method on logical key hierarchy for secure broadcast transmission. In: 2017 9th International conference on computational intelligence and communication networks
20. Deepa N, Pandiarajan P, Hybrid context-aware recommendation system for E-Health Care by Merkle hash tree from cloud using evolutionary algorithm. *Soft Comput*. <https://doi.org/10.1007/s00500-019-04322-7>
21. Sung S, Youn C, Kon E, Ryou J (2015) User authentication using mobile phones for mobile payment. In: 2015 IEEE
22. Saranya A, Naresh R (2021) Efficient mobile security for E health care application in cloud for secure payment using key distribution. In: Neural processing letters. Springer, Berlin. <https://doi.org/10.1007/s11063-021-10482-1>
23. Saranya A, Naresh R (2021) Cloud based efficient authentication for mobile payments using key distribution method. *J Ambient Intell Hum Comput*. <https://doi.org/10.1007/s12652-020-02765-7>
24. Naresh R, Vijayakumar P, Jegatha Deborah L, Sivakumar R (2020) “A novel trust model for secure group communication in distributed computing”, special issue for security and privacy in cloud computing. *J Organ End User Comput, IGI Global* 32(3)
25. Naresh R, Sayeekumar M, Karthick GM, Supraja P (2019) Attribute-based hierarchical file encryption for efficient retrieval of files by DV index tree from cloud using crossover genetic algorithm. *Soft Comput* 23(8):2561–2574 (Springer)
26. Naresh R, Gupta A, Sanghamitra (2020) Malicious url detection system using combined svm and logistic regression model. *Int J Adv Res Eng Technol (IJARET)* 10(4):63–73
27. Meenakshi M, Naresh R, Pradeep S (2019) Smart home: security and acuteness in automation of IOT sensors. *Int J Innovative Technol Exploring Eng (IJITEE)* 9(1):3271–3274

Extractive Text Summarization Using Feature-Based Unsupervised RBM Method



Grishma Sharma, Subhashini Gupta, and Deepak Sharma 

Abstract A methodology for creating shorter and meaningful summaries for single documents is provided. With a lot of content to be had on the web, it's far simply no longer possible to go through each information source in complete detail. Consequently, a great mechanism is needed to extract relevant information. To overcome these challenges, information in the form of text is summarized with the objective to get relevant knowledge without loss of any information. A methodology for extractive text summarization for single-document summary is devised and developed in this work. It uses a restricted Boltzmann machine to choose essential phrases from the text. The text documents used for summarization are in the English language. Various aspects are used to generate meaningful phrases, and the restricted Boltzmann machine is being utilized to enrich and abstract those features to improve the consequent accuracy without sacrificing any significant information. The sentences are scored, and an extracted summary is created based on those enhanced features. The result indicates that the presented methodology tackles the problem of text overload by producing an appropriate summary. The result of RBM has been compared with the Text Rank, Lex Rank, LSA, and Luhn algorithm. The experimentation is carried out, and the summary is generated for eight different document sets and the result is evaluated using the ROUGE-1 score.

Keywords Extractive text summarization · RBM · Feature extraction

G. Sharma (✉) · S. Gupta · D. Sharma
Department of Computer Science, K. J. Somaiya College of Engineering, Mumbai, India
e-mail: neelamotwani@somaiya.edu

S. Gupta
e-mail: subhashini.g@somaiya.edu

D. Sharma
e-mail: deepaksharma@somaiya.edu

1 Introduction

The problem of summarizing text is very trivial. In this fast-paced life, everyone wants shorthand information to save time. People read the news by only reading the headlines and the first few lines. A popular app that is exploiting this nature of humans is “Inshorts”, which provides a news summary in just 60 words. Students want a summary of class notes just one night before the exams. They also want a summary of YouTube lectures that may be hours long. NLP has proven to be very useful to solve this problem. While extractive summarization is popular currently, abstractive summarization is picking up pace. With the ever-rising amount of data in the globe, the demand for automatic summary generation from the text document is growing considerably to reduce the manual work of a person. In today’s world, data generation and consumption are exploding at an exponential rate. Text summarization finds its applications in various NLP-related tasks such as question answering, text classification, and other related fields. Summaries are generated as an intermediary step in these systems, which help to reduce the length of documents. This, in turn, leads to faster access for information searching. News summarization and headline generation is another important application. Most of the search engines use machine-generated headlines for displaying news articles in feeds.

The focus of this research is on extractive text summarization that aims to implement a single-document summarizer system that achieves a quick, concise extractive summary of any textual document. The structure of paper is as follows: Sect. 1 is introduction, Sect. 2 presents literature survey, the proposed methodology is given in Sect. 3, the results and discussion are in Sect. 4, and finally, the conclusion and future aspects are in Sect. 5.

2 Literature Survey

Madhuri and Ganesh Kumar [1], a unique statistical methodology is proposed and proven for extractive text summarization on a single document. The approach is described to extract sentences of the input text in a concise fashion. Weights are assigned to sentences to rank them. From the input text, highly ranked sentences are extracted. They created a summarizer application that accepts text files as input. After that, the input file is pre-processing. The system then calculates the sum of weighted frequencies by dividing the frequency of the keywords by their greatest frequency. Finally, the summarizer extracts high-weighted-frequency sentences and converts them to audio.

Krishnaveni and Balasundaram [2] proposed strategy for improving the summary coherence, based on local scoring and ranking. The author creates two types of summaries in this paper: heading-wise and main. Features are usually employed for the score of phrases. The original text is separated heading by heading, and each heading is treated as a separate text.

Naik and Gaonkar [3], the rule-based concept was proposed to extract features from sentences. The author pre-processed the input data first and then retrieved keywords from the document, which were subsequently pruned based on the computed threshold. Then the feature value is calculated for each document. Certain rules have been written by the author. Finally, the sentence is sorted based on sentence score to form an extractive summary.

Jafari and Shahabi [4], presented a fuzzy logic method to calculate and encode feature values as feature attribute vectors for each text. The fuzzy system assigns a score between 0 and 1. Input values for membership function are calculated. The knowledge base also contains the rules needed for summarization. Finally, top n scored sentences are selected to form a summary.

One of the most well-known extractive text summarizing algorithms is stated by Luhn [5], and it is determined by the number of times words appear in the text, as well as the distance between relevant words, which is determined by the number of non-relevant words among relevant ones.

The text rank algorithm is generally based on the graph algorithm which is developed by Mihalcea and Tarau [6]. The text rank algorithm works in two steps, very first it calculates the similarity between two sentences, and in the second step, it calculates the overall significance of sentences.

3 Proposed Methodology

3.1 Data Pre-processing

1. *Sentence Segmentation*: Segmentation breaks down the entire text into sentences, once the sentences are broken then each sentence with their respective position will be stored in the array.
2. *Tokenization*: This data pre-processing step sentences are divided into tokens so that they can be further used for feature extraction tasks.
3. *Stop word and punctuation*: In this data pre-processing task, we will remove punctuation as well as often occurring words such as punctuation, but, the, a, and so on. Proposed methodology for extractive text summarization is shown in Fig. 1.

3.2 Feature Extraction

1. *Sentence Position*: The position of the sentence can determine the importance of the sentence for the summary. The sentences at the beginning and end of the document are usually the most significant. So, based on this the sentence score is



Fig. 1 Proposed methodology

calculated. In our case, the positional score is determined by taking into account the following factors:

$$\text{Sentence_Position} = \begin{cases} 0, & \text{if it's first or last sentence of text} \\ \text{else } \cos(\text{Sen_pos} - \min)((1/\text{max}) - \min) \end{cases}$$

where,

Sen_pos: Position of sentence in text. Min: th * total no. of sentence in the text. Max: th*2* total no. of sentence in the text. Th: threshold which to be considered as 0.2.

2. *Tokenization*: This feature helps to filter out too short sentences, which typically don't convey much information.

$$\text{Sentence_Length} = \begin{cases} 0, & \text{if number of words is less than 3} \\ \text{else number of words in the sentence} \end{cases}$$

3. *Proper nouns*: A proper noun refers to something with a particular identity, such as a name or a location. In this first sentences are tagged using POS tagger using NLTK Library. Then we count the proper noun in tagged sentence.

$$\text{proper_noun} = \left\{ \frac{\text{numberofpropernounintagged } S_i}{\text{totallengthoftagged } S_i} \right.$$

where

s_i : i th sentence. Number of proper noun in tagged s_i : number of proper noun in i th sentence. Total length of s_i : number of words in i th sentence.

4. *Number of numerals*: Since the numerals in a document represent facts, having sentences with specific figures is important. The number of numerical can be calculated by using the following formula.

$$\text{number_of_numericals} = \left\{ \frac{\text{numberofnumericalsin } S_i}{\text{totalnumberofwordsin } S_i} \right.$$

5. *No. of Thematic words*: Thematic terms are the top ten most commonly used words in the sentence. The no. of thematic terms can be calculated as follows.

$$\text{thematic_words} = \frac{\text{numberofthematicwordsin } S_i}{\text{totalnumberofthematicwords}}$$

6. *Centroid similarity*: The centroid sentence is defined as the sentence with the maximum TF-ISF score. The cosine similarity of each sentence to the centroid sentence will then be computed.

$$\text{Centroid_Similarity} = \{\text{cosine_similarity}(\text{centroid}, \text{sentence})\}$$

7. *TF-ISF*: TF-ISF represents Term frequency—Inverse document frequency that’s work similar to TF-IDF works. TF_ISF is given as follow:

$$\text{TF_ISF} = \left\{ \frac{\sum \log(\text{isf}) \times \text{tf}}{\text{Total words}} \right.$$

8. *Named entities*: In this feature extraction section, we calculate the number of the named entity in every sentence. Sentences include references to named individuals such as a corporation, a group of people, and so on are often required to understand a factual report in some way.

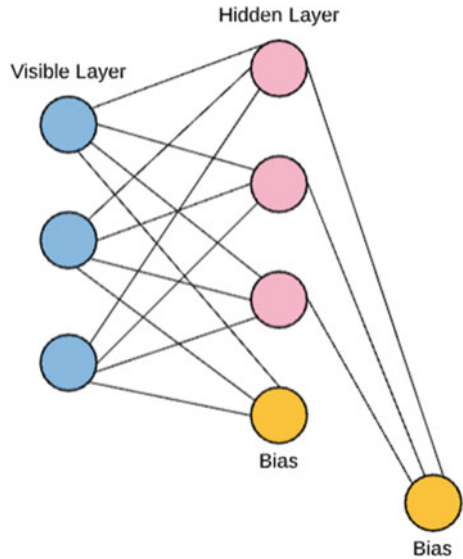
Once the feature value of each sentence is obtained, we will generate the sentence feature matrix of sentence. Sentence feature Matrix is a two-dimensional matrix as shown in Fig. 2. Sentence feature matrix $S = (s_1, s_2, s_3, s_4, \dots, s_n)$ where $s_i = (f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8)$ is a sentence feature and $(i \leq n)$. Where total no. of sentences in the document is indicated by n .

The sentence feature matrix produced by the preceding stages is as follows:

Fig. 2 Feature matrix for text summarization

$$\begin{matrix} s_1 \\ s_2 \\ \cdot \\ \cdot \\ s_n \end{matrix} \begin{bmatrix} f_1 & f_2 & f_3 & f_4 & f_5 & f_6 & f_7 & f_8 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix}$$

Fig. 3 RBM model

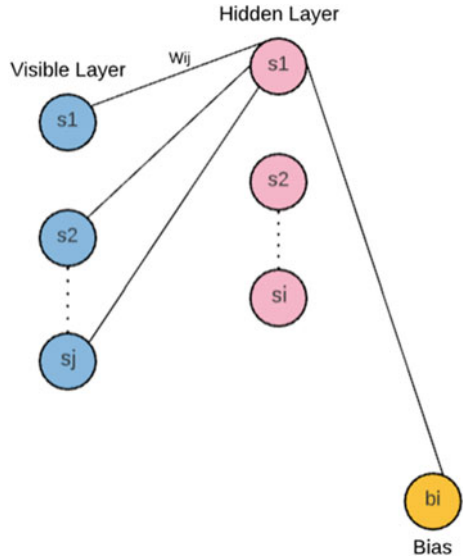


3.3 *Restricted Boltzmann Machine*

Our approach of extractive summarization is restricted Boltzmann machine (RBM), a stochastic and generative neural network that is capable of learning internal representations through probability distribution over its set of inputs [7]. They are a two-layered artificial neural network; the first layer is called the visible layer or the input layer (input nodes), and another one is called the hidden layer (hidden nodes). Every hidden node is connected to the bias node. The input nodes are not related to one another in the visible layer. Also, in the hidden layer, hidden nodes are not related to one another. The network is known as the restricted Boltzmann machine because of these restricted connections (Fig. 3).

To improve, the feature matrix S is fed into an RBM with one hidden layer. Each sentence passes through hidden layer 1 initially. Each sentence’s feature values are multiplied by randomly produced weights, and one bias value is randomly produced and added to all the sentences. The results of these operation are fed into an activation function, which produces the nodes output (Fig. 4).

Fig. 4 Visible layer to Hidden layer



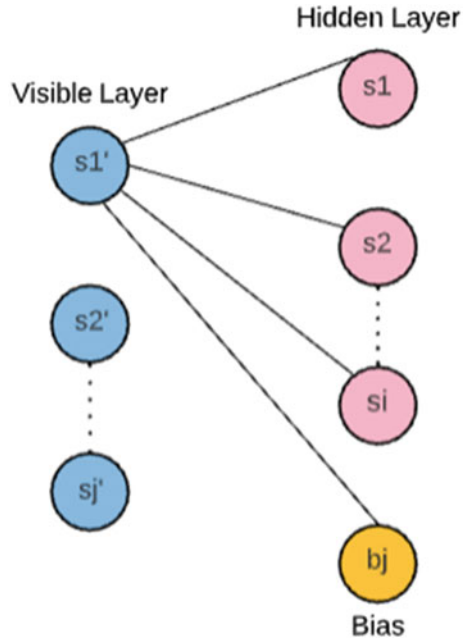
$$P(s_i) = \sigma \left(\sum_{j=1}^n s_j \times W_{ij} + b_i \right)$$

$$\sigma(x) = \frac{1}{(1 + e^{-x})}$$

where, $\sigma(x)$ is the sigmoid function, w_{ij} stands for randomly generated weights, whereas b_i stands for bias.

After this, the restricted Boltzmann machine learns to reconstruct data by itself in an unsupervised manner. This is done by reversing the above process, i.e., the hidden layer will become the input layer with activations as the new input. Then these activations are again multiplied with previous weights which are associated with the visible layer nodes and these products are added to the visible layer bias at each visible node. Therefore, obtained results are known as the reconstructions which are then compared to the original input (Fig. 5).

Fig. 5 Hidden layer to visible layer



The likelihood of activation of a visible unit s_j is stated as

$$P(s_j) = \sigma \left(\sum_{j=1}^n s_j \times W_{ij} + b_i \right)$$

$$\sigma(x) = \frac{1}{(1 + e^{-x})}$$

where, $\sigma(x)$ is the sigmoid function, w_{ij} is weights and b_i is the bias.

Thus, the hidden unit values are predicted during training and after determining the hidden unit values, the new input values are predicted. This procedure is known as Gibbs sampling. The training loss is obtained by calculating the difference between the old s'_j and new input values s_j . This procedure is performed for a number of epochs, and the weights are determined using (CD) contrastive divergence given by:

$$w_{ij(\text{new})} = w_{ij(\text{old})} + (\text{Learning_Rate} \times w')$$

$$w' = (s_j \otimes P(s_i) - s'_j)$$

where, w' is the difference between the outer products of probabilities with the original input values s_j and the new input values as a change in weights values s'_j . An enhanced feature matrix is obtained after the epochs which have enhanced feature values for each sentence.

3.4 Summary Generation

A list is created that contains the sum of all enhanced feature values for each sentence in the document. As a result, a value is generated for each sentence, which is the sentence’s score. Sentences are sorted according to their scores in decreasing order. The first sentence is always included in the summary because it is the most essential sentence. The top 50% of the remaining sentences are included in the first summary and sorted in descending order according to their original location in the text.

4 Result and Discussion

We have used the BCC news dataset which includes several articles from different domains such as technology, politics, business, entertainment, and sport along with human-generated summaries for those articles. We have used ten documents from the BCC new dataset for evaluation purpose. Proposed methodology is compared with other state of art extractive summarization algorithm such as Text Rank [6], Lex Rank [8], LSA [9], Luhn [5]. For generating summary from Text Rank, Lex Rank, LSA, Luhn we have used Sumy Tool. By using Sumy tool, we can directly import this entire algorithm and for creating summary by using these algorithm users have just pass the text document. Each document is summarized with four different summarization techniques. We have extracted the top-ranked sentence from the document to form a summary at 10%, 20%, 30%, 40%, and 50% of document length. The purpose of experimenting with different percentage levels is to investigate the performance of various approaches.

We evaluated each system summary in conjunction with its corresponding reference summary. The ROUGE tool was used for the evaluation. ROUGE includes precision, recall and f-measure. The result fshows that proposed approach give better result than all other techniques that stated above (Tables 1, 2, and 3).

Table 1 Precision

Technique	Summary length (%)				
	10%	20%	30%	40%	50%
Text Rank	0.54	0.54	0.48	0.47	0.49
Lex rank	0.60	0.50	0.58	0.48	0.41
LSA	0.54	0.63	0.58	0.47	0.42
Luhn	0.75	0.62	0.55	0.51	0.44
RBM	0.87	0.74	0.70	0.52	0.54

Table 2 Recall

Technique	Summary length (%)				
	10%	20%	30%	40%	50%
Text Rank	0.27	0.44	0.46	0.60	0.75
Lex Rank	0.19	0.38	0.52	0.62	0.73
LSA	0.25	0.43	0.50	0.61	0.63
Luhn	0.25	0.35	0.41	0.52	0.56
RBM	0.30	0.45	0.56	0.66	0.82

Table 3 F1 Score

Techniques	Summary length (%)				
	10%	20%	30%	40%	50%
Text rank	0.36	0.48	0.47	0.51	0.66
Lex rank	0.33	0.52	0.50	0.40	0.59
LSA	0.34	0.54	0.58	0.53	0.68
Luhn	0.38	0.45	0.47	0.52	0.49
RBM	0.40	0.56	0.62	0.58	0.75

5 Conclusion

We have developed an unsupervised methodology that makes use of RBM to summarize single document. The algorithm works separately for each input document, as each document is uniquely different in itself. This is an advantage that the proposed methodology gives. Our method uses RBM and generates an effective and efficient summary. We have evaluated our model using the ROUGE1 score compared with other techniques, and the result shows that our model gives better results than other compared techniques.

The proposed methodology could be extended to multiple-document summarization. Different languages can be used to summarize documents.

References

1. Madhuri JN, Ganesh Kumar R (2019) Extractive text summarization using sentence ranking. In: IEEE international conference on data science and communication, Mar 2019
2. Naik SS, Gaonkar MN (2017) Extractive text summarization by feature-based sentence extraction using rule-based concept. In: IEEE international conference on recent trends in electronics information and communication technology (RTEICT), May 2017
3. Krishnaveni P, Balasundaram SR (2016) Automatic text summarization by local scoring and ranking for improving coherence. In: IEEE international international conference on automation and computing, Sept 2016

4. Jafari M, Shahabi AS, Wang J, Qin Y (2017) Automatic text summarization using fuzzy inference. In: Proceedings 22nd international conference on automation and computing, May 2017
5. Luhn HP (1958) The automatic creation of literature abstracts. *IBM J Res Dev* 2
6. Mihalcea R, Tarau P (2004) Text rank: bringing order into texts. In: Association for Computational Linguistics, July 2004
7. Sharma B, Tomer M, Kriti K (2020) Extractive text summarization using F-RBM. *J Stat Manage Syst* 23(6)
8. Erkan G, Radev DR (2004) Lex rank: graph-based lexical centrality as salience in text summarization. *J Artif Intell Res*
9. Xiangen H, Zhiqiang C, Max L, Andrew O, Phanni P, Art G (2003) A revised algorithm for latent semantic analysis. In: Proceedings of the 18th international joint conference on artificial intelligence

Depression and Suicide Prediction Using Natural Language Processing and Machine Learning



Harnain Kour and Manoj Kumar Gupta

Abstract Depression has always been one of the prominent concerns of mental health worldwide. In the worst-case scenario, someone suffering from depression may lead to drastic measures such as suicide. According to the World Health Organization, depression and anxiety affect one out of every five people worldwide, costing trillions of dollars each year. In the COVID-19 pandemic, the situation has worsened alarmingly as more people suffer from depression. It has become essential, more than ever, to maintain the mental health profiles of our people and to predict any unfortunate event. Depression can be prevented and treated at a very early stage and a low cost, given early detection and identification of the causes. With advancements in machine and deep learning models, it has become possible to identify such behaviour through social interactions such as posts, tweets, and comments. This paper aims to detect user behaviour that can conclude whether a person is suffering from depression and suicidal tendencies based on the user's social media tweets. The research work proposes a classifier with a hybrid technique in preprocessing using Natural Language Processing (NLP) and machine learning techniques with an accuracy of 75% to identify such traits in a person through his/her tweets.

Keywords Health · Depression · Sentiment analysis · Social platforms · Machine learning · Natural language processing

1 Introduction

The well-being of a person comprises physical health and mental health. The mental health of a person shows the individual's state of mind. With the advancement in technology and social interactions, a cultural shift has been observed in terms of social behaviour, moods, lack of happiness, etc. It has contributed significantly to

H. Kour (✉) · M. K. Gupta
School of Computer Science and Engineering, SMVDU, Katra, Jammu & Kashmir, India
e-mail: 18dcs008@smvdu.ac.in

M. K. Gupta
e-mail: manoj.gupta@smvdu.ac.in

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
D. P. Agrawal et al. (eds.), *Cyber Security, Privacy and Networking*, Lecture Notes in Networks and Systems 370, https://doi.org/10.1007/978-981-16-8664-1_11

117

increase hopelessness, a lack of external contact, and other factors that eventually lead to depression or suicidal ideation [1]. People ought to communicate and express themselves on social media through multiple channels. Occasionally they either make a conversation with others or express themselves by posting on the platform. Twitter provides a platform where users share their thoughts, emotions, feelings, and expressions. These tweets can aid in determining a person's thought process, mental health, and behavioural traits [2]. The dangerous mental condition usually gets reflected in the person's social behaviours, such as tweets. While some patients with depression are more likely to be diagnosed accurately, others rarely experience/express the symptoms. Depression has become a silent killer, with no visible symptoms or identification, and it must be addressed as soon as possible [3]. Identification and detection of mental health issues have been at the forefront of ongoing medical research worldwide. AI has changed the complete landscape of identifying extreme behaviours like depression, severe obsessive-compulsive disorder, addiction to drugs, mental stress, anxiety, and personality disorders [4].

This research work focuses on the potential of NLP and machine learning techniques that can be utilized in the mental health field. NLP has been instrumental in understanding the context of natural human language. As a result, it extracts latent meaning from text and creates AI-based solutions using massive amounts of text data available on social media platforms, the internet, and other sources [5]. The objective is to come up with a methodology to accurately classify a user's tweet implying whether he is suffering from depression or not. The paper continues with a research background in Sect. 2 and will discuss some of the existing work done in the mental health domain using machine learning and NLP techniques giving a brief overview of the research challenges. Section 3 discusses the dataset and the preprocessing methodology used to prepare the entire dataset before training the model. Section 4 delves into the technical methods used and the motivation for selecting techniques, model training, and architecture. Section 5 discusses the experimental results with supporting tables and figures. Finally, in Sect. 6, we wrap up the paper and make some recommendations for future improvements.

2 Related Work

This study was carried out on social media data to predict depression and suicidal ideation because there has been little research in this field. By observing the behaviour of people suffering from depression, one can prevent them from taking harmful actions or committing suicide and help them with the assistance of friends, family, and psychologists. As it is a naive field, a great deal of research still needs to be carried out. This section discusses some of the work done earlier in this area. Leiva et al. [6] investigated clinical trials that aid in predicting depression risk. For a specified time, the author extracted messages posted by online users. The study combined a few machine learning techniques, such as KNN, random forest, SVM, and the ensemble learning algorithm, to detect early symptoms of depression from

social media without delay. Text features were extracted using TF-IDF vectorization, and vector dimensionality was reduced using PCA. In addition, the author investigated genetic algorithms and text polarity and discovered that they improved previous results by 16.7% compared to the baseline. Zheng et al. [7] proposed a model to address the weak points of deep learning-based models. Using nodes from knowledge graphs, the authors developed a graph attention system. A knowledge graph mechanism is used to improve the system's classification performance. The real-time datasets demonstrate that the proposed method fundamentally improves order and forecast execution when compared to other significant best practices. The macro precision, recall rate, and F1-measure were used to evaluate the final results. The system achieved a 95.4% F1 measure, a precision rate of 93.0%, and a recall rate of 98.0%. Tong et al. [8] envisioned a model that would be useful for classifying depression in online users and developed a novel classifier, specifically a boosting method, to classify non-depressed and depressed Twitter users. The accuracy obtained on the Twitter Dataset was 91.54, the Glass Dataset was 95.85, the LSVT Dataset was 89.48, and the News Dataset was 78.0. Pete Burnap et al. [9] had a methodology to classify tweets within multiclass classification related to communication converging to suicidal tendencies. To identify terms commonly used in the class of suicidal ideation, they used Term Frequency (TF)—Inverse Document Frequency (IDF), which is a numerical embedding of text. Another dimensionality reduction technique implemented was PCA (Principal Components Analysis). Three machine classifiers have been used, i.e. SVM, Naïve Bayes. SVM, Decision Tree, and the algorithm did work well with the short informal text. In addition, an ensemble classification approach was incorporated to refinish the more accurate classification. The technique of the ensemble combined the feature sampling method and used a basic classification for the training phase. The rotation approach of the forest ensemble improved performance. The best classifier was SVM. Bart Desmet et al. [10] introduced text classification-based methods that automatically discover online content related to suicides in the Dutch language from some forum posts. Two prominent algorithms prominently used for text classification are SVM and Naïve Bayes. Other advanced algorithms, such as genetic algorithms, are used for parameter optimization and the best features and hyperparameters [11] for the generation of the SVM model. The F1-score is the metric selected for model evaluation. After joint optimization, best models achieve an F1-score of 92.69%. The highest F1-score attained was 69.51% and is acquired by the technique of stratified feature group selection. Hiraga et al. [12] used traditional machine learning algorithms such as multinomial logistic regression, Naive Bayes, and linear support vector machines to classify mental disorders from Japanese blogs. Wu et al. [13] proposed various hypotheses and their correlations based on language, time, and interaction to predict job burnout on 1532 Weibo burnout users using machine algorithms like decision trees, logistic regression, support vector machines, XGBoost, and random forests which replaced previous statistical methods based on surveys. Discovering various medical and biomedical lapses is what soft computing for medical intelligence is all about determining the correlation between various aspects of data. Wallison et al. [14] inundate on the genomic briefing of the SARs virus. On further reading for lung

tumours [15] and oxygen toxins [16], one could conclude that the medical diagnosis of various immunological responses and attacks has been majorly recorded under the concept of computer vision as a soft computing technique. In this paper, we however try to define the abrupt ability of natural translation techniques to feature extract the possibility of depression from the patient's tweet or any written record. This is because often depression patients are not aware of their illness; thus, this unsupervised work which was earlier a full-time medical consultation with stigma and hesitation attached to it. We take inspiration and ideas from Clarizo et al. [17], who demonstrated similar sentiments on social networks using traditional neural networks. Similarly, Casillo et al. [18] demonstrate the usage of the teller bot algorithm of touristic activities such that the sentiment of the touristic attraction with abstraction is detected. Along with similar lines, Colace et al. [19] tried to demonstrate another useful cycle bot usage in London creates a feature map and tries to convolute stimulation on stigma-based sentiment analysis.

2.1 Challenges

As depression has no evident symptom, its recognition becomes a challenge to identify its user behaviour pattern. Artificial intelligence/ML/DL provides end-to-end solutions and has proven to be a highly effective tool in detecting such patterns. The machine learning technique has been found to be accurate in text classification when the context of the text is clear, straightforward, and formal, and there is a clear distinction between whether it is depressive or not. However, a person expresses his/her feeling in complex and layered ways, and thus using text classification on such text with latent meaning is still a complicated task. There can be multiple intents behind the same tweet as teenagers can tweet jokes sounding like a severe life concern, while a person with mental health issue can express his true statement in a twisted tweet. Also, many users do not openly share things about their depression online. So, the model must be robust enough to capture the relationship between words that capture characteristics of a depressed person's tweets.

The research work proposes a solution that follows a set of steps to detect depression or suicidal tendencies. Initially, we start with extracting the user tweets from Twitter. It will provide high-quality text data of human expressions, including both normal and depressive tweets. Following that, we perform the necessary preprocessing and data cleaning on the raw extracted text. Next, we will build the first part of the process by transforming the text data into the numerical format using the appropriate methodology. Some of such methods are TF-IDF (Term Frequency-Inverse Document Frequency) vectorization, word2vec, Fast-Text word embeddings, etc. For this study, we used a hybrid embedding technique of TF-IDF and fastText to provide enhanced text embedding. We chose appropriate classifiers to categorize these tweets as 'Depression,' 'Suicidal,' and 'Teenager (normal)'. The classifier was chosen based on its robustness and underlined technique. We have selected an ensemble classifier, random forest, and one margin classifier, support vector machine (SVM).

To further optimize the classifier performance, we have tuned the model parameters. Ultimately, we have compared the SVM and random forest models generated against the selected metrics, accuracy, recall.

3 Dataset Description and Processing

This study’s dataset was obtained from Kaggle and consists of 349330 tweets divided into training and test sets [20]. There are three categories of tweets: depression, suicide, or normal tweets. The training set and test set have been created from the above dataset with random shuffling and stratified splitting in the ratio of 75:25, respectively.

3.1 Dataset Preprocessing

The dataset contains raw and unclean text. For modelling purposes, we require clean and relevant text [21]. Hence, detailed data preprocessing and cleaning are required for the machine learning algorithm to learn correct text sequences and classify them.

Data preprocessing steps are shown in Fig. 1. Raw data is processed by lowering the entire corpus and using NLP techniques such as stemming or lemmatizing, converting the words into a single representation, and normalizing the words into root form or base form. The most frequent words called stop words and special symbols that do not add value to text data need removal. Few words with fewer occurrences, i.e. less than five times in the complete corpus, hold no value for the training, and thus these are also filtered out [22].

We need to convert the processed data into a compatible format. Tokenization, an NLP technique that converts the raw text of a paragraph or sentence into individual words, is used. It treats each word as a separate unit that can be easily converted into a numerical format and used as input to machine learning algorithms [23].

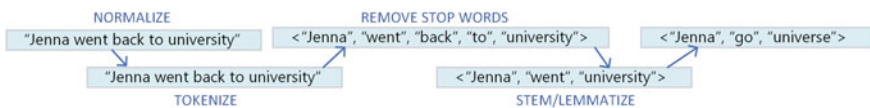


Fig. 1 Tweets preprocessing steps

4 Methodology

From the literature survey, we found that ensemble classifiers are proven to be one of the best techniques for classifying text data. Hence, a random forest classifier is selected for this purpose. The other model we choose is a support vector machine with a performance comparable with random forest but uses the support vector technique for classification. Figure 2 below shows the complete end-to-end pipeline implemented with the constituent techniques used at each phase.

For the exploratory data analysis purpose, we have employed two techniques: word-cloud [24] and frequency distribution plot [25]. Figure 3 depicts a word-cloud of the most important and frequently used words by users in tweets labelled as depressed, while Fig. 4 depicts frequency counts of the same words.

Once the data has been cleaned and is ready for further processing, we choose the representation methodology for the words and tweets, ranging from basic techniques like TF-IDF to advanced word2vec and fastText-based embeddings.

We have used a hybrid solution to supercharge the word vectors for our approach, combining both TF-IDF and fastText embeddings to represent the individual tweets. In our work, this representation technique is superior to both constituent techniques when implemented separately. Supercharged vectors can be helpful as they reduce the overall dimensionality of data but capture individual words' importance. Having a hybrid embedding representation of individual tweets only solves half the problem statement. The essential parameters for algorithm selection are the data size, the number of classes to classify, and underlying techniques. For benchmarking and comparing the results, we selected the most basic algorithms for the multiclass classification task. Deep learning classifiers are feasible where the training data is too

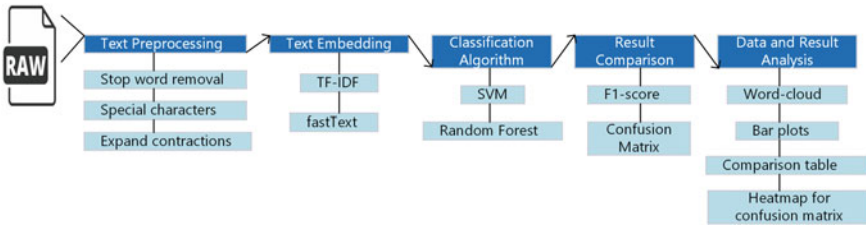


Fig. 2 Flow of methodology and techniques used at each phase

Fig. 3 Word-cloud for tweets labelled as depressed



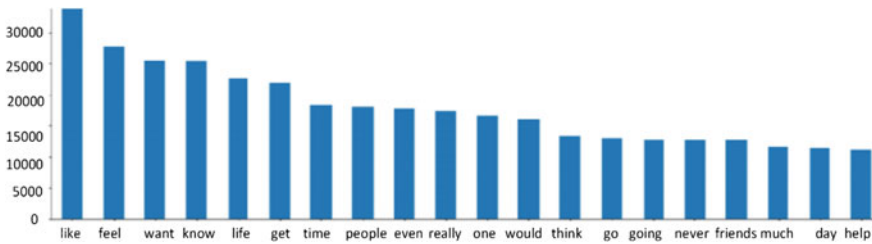


Fig. 4 Word frequency plot

large and the class space is too high. With only three classes to classify, we can achieve similar or maybe better results with our data.

The TF-IDF [26] is a numerical statistic technique that calculates the importance of individual words within a document in a corpus. In information retrieval, it is used as a weighting factor [27]. As the frequency of the word increases in a document, the TF-IDF value increases but inversely depends on the frequency of the word in the corpus. Therefore, if the words appearing in the corpus are rare, then they are weighted high. fastText is an embedding technique that can create a numerical representation of words or complete sentences [28]. fastText is trained on character n-grams using deep learning techniques and provides universal embedding for out-of-the-word vocabulary. For our implementation, we will train the embedding vectors on our cleaned dataset for ten iterations and a context window of size 3. The following steps summarize the hybrid technique using the above two methods:

1. Calculation of TF-IDF vector for each word and, therefore, each tweet in the corpus.
2. Find the individual word embeddings using fastText pre-trained embeddings.
3. Multiply the individual word embeddings with the TF-IDF weights of the corresponding words.
4. To find the representation of the tweets, take the average of all the word embeddings of a word in the tweet.

4.1 Machine Learning Classifiers

We used the two most common algorithms for the analysis but with different underlying principles for classification. SVM [29] uses support vectors or decision boundaries, whereas RF [30] is an ensemble learning technique and is a tree-based classifier.

1. **SVM:** SVM creates a classification boundary using kernel functions to approximate complex decision boundaries.
2. **Random Forest:** This is an advanced ensemble technique [31] with decision trees as its underlying classification algorithm. It is a robust method and can give state-of-the-art results too.

Hyperparameter Optimization: The SVM hyperparameters are optimized manually for performance. The specifics are as follows: C = '1.0', Gamma = 'auto', and the maximum depth of the tree for Random Forest = 10 to avoid overfitting the model.

5 Results and Experiments

For every algorithm selected for classification, the final results are compared across the five metrics to identify the best algorithm for the dataset used. The classification metrics [32] used to compare and evaluate the results quantitatively is as follows:

- **Confusion Metrics:** A complete matrix to calculate the TP (True Positive), FP (False Positive), TN (True Negative), and FN (False Negative) and confusion matrix for SVM and random forest is shown in Figs. 5 and 6.
- **Precision:** It refers to the correct prediction of the numbers of the total positive values and is shown in Eq. (1):

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \tag{1}$$

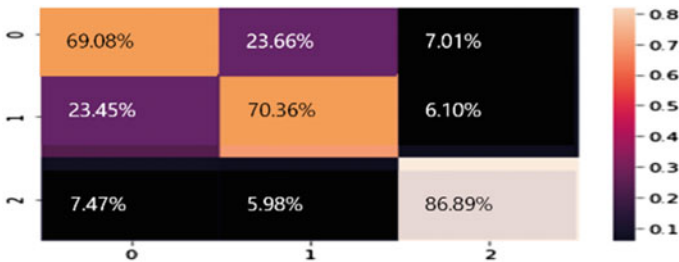


Fig. 5 Confusion matrix for SVM

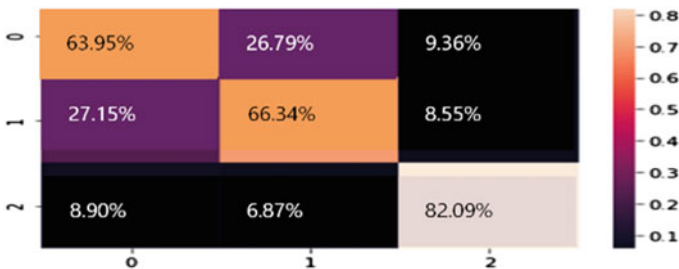


Fig. 6 Confusion matrix for RF

Fig. 7 Recall comparison of SVM

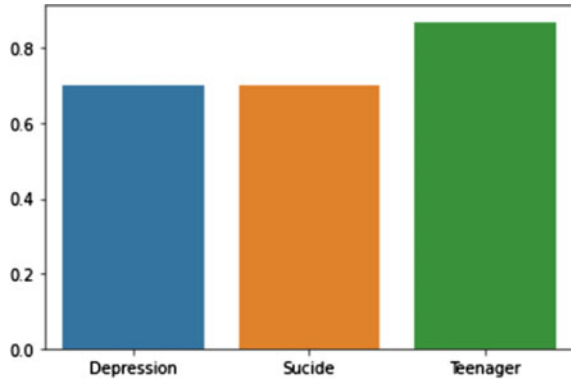
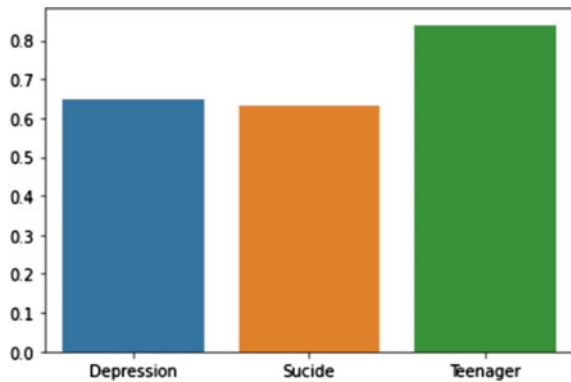


Fig. 8 Recall comparison of RF



- **Recall:** It refers to how much of the actual positive value we predicted correctly and is shown in Eq. (2) and comparison of recall for SVM and random forest is shown in Figs. 7 and 8.

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN}) \tag{2}$$

- **Accuracy:** It refers to the total correctly predicted values across all classes and is shown in Eq. (3):

$$\text{Accuracy} = \frac{(\text{TP} + \text{TN})}{(\text{TP} + \text{FP} + \text{TN} + \text{FN})} \tag{3}$$

- **F1-Score:** It refers to the harmonic mean of recall and precision or a more robust score than either precision or recall. Table 1 shows the comparison of all performance parameters for SVM and RF on training and test data.

Table 1 Performance parameters comparison

Embedded technique	Classifier	Training metrics		Training data = 75%		Test metrics		Test data = 25%	
		Recall	Precision	F1-score	Accuracy	Recall	Precision	F1-score	Accuracy
TF-IDF + fastText	SVM	71	70	71	77	70	69	69	75
	RF	72	69	70	76	65	64	65	71

6 Conclusion

In this study, the outcomes show that even the basic classification algorithms on the dataset can achieve good accuracy and recall scores. However, there is a scope of improvement in both model and data representation techniques. Identifying tweets containing depression and suicidal content is not a difficult task if the pattern of the text is accurately captured and a clear distinction from other behaviour is made. We compared the results from two classification algorithms, SVM and random forest, with our experiments. Random forest is a tree-based ensemble algorithm, and it performed better than SVM with training data, but we prune the tree for appropriate depth to avoid overfitting the model. The approximate accuracy achieved for depression prediction by SVM is 75% and by random forest is 71%. In the future, an advanced and contextual deep learning-based classification model can be used to capture the long-term contextual relationship between the words in a tweet. The embedding or representation methodology can also be improved by utilizing state-of-the-art techniques such as Glove and BERT, which are true bidirectional word embeddings.

References

1. Beard C, Millner AJ, Forgeard MJ, Fried EI, Hsu KJ, Treadway MT, Björgvinsson T (2016) Network analysis of depression and anxiety symptom relationships in a psychiatric sample. *Psychol Med* 46:3359–3369
2. Lin C, Hu P, Su H, Li S, Mei J, Zhou J, Leung H (2020) Sensemood: depression detection on social media. In: *Proceedings of the 2020 international conference on multimedia retrieval*, pp 407–411
3. Coppersmith G, Dredze M, Harman C (2014) Quantifying mental health signals in Twitter. In: *Proceedings of the workshop on computational linguistics and clinical psychology: from linguistic signal to clinical reality*, pp 51–60
4. Kim HJ, Park SB, Jo GS (2014) Affective social network–happiness inducing social media platform. *Multim Tools Appl* 68(2):355–374
5. Kim K, Moon J, Oh U (2020) Analysis and recognition of depressive emotion through NLP and machine learning. *J Conv Cult Technol* 6(2):449–454
6. Leiva V, Freire A (2017) Towards suicide prevention: early detection of depression on social media. In: *International conference on internet science*. Springer, Cham, pp 428–436

7. Liu J, Zheng Y, Dong K, Yu H, Zhou J, Jiang Y, Ding R (2020) Classification of fashion article images based on improved random forest and VGG-IE algorithm. *Int J Pattern Recogn Artif Intell* 34:2051004
8. Tong L, Zhang Q, Sadka A, Li L, Zhou H (2019) Inverse boosting pruning trees for depression detection on Twitter. arXiv preprint [arXiv:1906.00398](https://arxiv.org/abs/1906.00398)
9. Burnap P, Colombo G, Amery R, Hodorog A, Scourfield J (2017) Multi-class machine classification of suiciderelated communication on twitter. *Online Soc Netw Media* 2:32–44
10. Desmet B, Hoste V (2018) Online suicide prevention through optimised text classification. *Inf Sci* 439:61–78
11. D'Angelo G, Palmieri F (2021) GGA: a modified genetic algorithm with gradient-based local search for solving constrained optimization problems. *Inf Sci* 547:136–162
12. Hiraga M (2017) Predicting depression for Japanese blog text. In: *Proceedings of ACL 2017, Student research workshop*, pp 107–113
13. Wu J, Ma J, Wang Y, Wang J (2021) Understanding and predicting the burst of burnout via social media. *Proc ACM Hum-Comput Inter* 4(CSCW3):1–27
14. D'Angelo G, Palmieri F (2020) Discovering genomic patterns in SARS-CoV-2 variants. *Int J Intell Syst* 35:1680–1698
15. Elia S, D'Angelo G, Palmieri F, Sorge R, Massoud R, Cortese C, De Stefano A (2019) A machine learning evolutionary algorithm-based formula to assess tumor markers and predict lung cancer in cytologically negative pleural effusions. *Soft Comput* 1–13
16. D'Angelo G, Pilla R, Dean JB, Rampone S (2018) Toward a soft computing-based correlation between oxygen toxicity seizures and hyperoxic hyperpnea. *Soft Comput* 22(6):2421–2427
17. Clarizia F, Colace F, Lombardi M, Pascale F, Santaniello D (2019) Sentiment analysis in social networks: a methodology based on the latent Dirichlet allocation approach. In: *Proceedings of the 11th conference of the European society for fuzzy logic and technology (EUSFLAT 2019)*, Prague, Czech Republic, pp 9–13
18. Casillo M, Clarizia F, D'Aniello G, De Santo M, Lombardi M, Santaniello D (2020) CHAT-Bot: a cultural heritage aware teller-bot for supporting touristic experiences. *Pattern Recogn Lett* 131:234–243
19. Colace F, De Santo M, Lombardi M, Pascale F, Santaniello D, Tucker A (2020) A multilevel graph approach for predicting bicycle usage in London area. In: *Fourth international congress on information and communication technology*. Springer, Singapore, pp 353–362
20. Suicide and depression detection using subreddit and reddit platform (online). <https://www.kaggle.com/nikhileswarkomati/suicide-watch>
21. Kumar Y, Sood K, Kaul S, Vasuja R (2020) Big data analytics and its benefits in healthcare. In: *Big data analytics in healthcare*. Springer, Cham, pp 3–21
22. Go A, Bhayani R, Huang L (2009) Twitter sentiment classification using distant supervision. *CS224N Project Report*, Stanford, 1(2009), p 12
23. Luo LX (2019) Network text sentiment analysis method combining LDA text representation and GRU-CNN. *Pers Ubiquitous Comput* 23(3):405–412
24. Heimerl F, Lohmann S, Lange S, Ertl T (2014) Word cloud explorer: text analytics based on word clouds. In: *2014 47th Hawaii international conference on system sciences*. IEEE, pp 1833–1842
25. Bullinaria JA, Levy JP (2007) Extracting semantic representations from word co-occurrence statistics: a computational study. *Behav Res Methods* 39(3):510–526
26. Qaiser S, Ali R (2018) Text mining: use of TF-IDF to examine the relevance of words to documents. *Int J Comput Appl* 181(1):25–29
27. Wu HC, Luk RWP, Wong KF, Kwok KL (2008) Interpreting tf-idf term weights as making relevance decisions. *ACM Trans Inf Syst (TOIS)* 26(3):1–37
28. Li Z, Xiong Z, Zhang Y, Liu C, Li K (2011) Fast text categorization using concise semantic analysis. *Pattern Recogn Lett* 32(3):441–448
29. Description of support vector machine algorithm. <https://towardsdatascience.com/support-vector-machine-introductionto-machine-learning-algorithms-934a444fca47>. Accessed 1 June 2021

30. Description of random forest algorithm. <https://towardsdatascience.com/the-random-forest-algorithmd457d499ffcd>. Accessed 29 May 2021
31. Gupta S, Gupta MK (2021) Computational prediction of cervical cancer diagnosis using ensemble-based classification algorithm. *Comput J* bxaa198
32. Kadhim AI (2019) Survey on supervised machine learning techniques for automatic text classification. *Artif Intell Rev* 52(1):273–292

Automatic Detection of Diabetic Retinopathy on the Edge



Zahid Maqsood and Manoj Kumar Gupta

Abstract The uncontrolled blood sugar levels in diabetes patients lead to an eye disease called diabetic retinopathy. The high sugar levels in the blood vessels of the retina cause blockage of some blood vessels due to which fluids like plasma leak easily into the eye causing the lesions which appear in the eye and may cause severe vision problems. A severe vision problem can be prevented by detecting and treating it at an early stage. In India alone, about 80 million people suffer from diabetes, and there is one ophthalmologist for every 100,000 population. Due to this serious shortage of well-trained ophthalmologists, it becomes difficult to diagnose the severity of diabetic retinopathy in some rural areas of India. Since most of the AI solutions for detecting diabetic retinopathy are cloud-based, therefore, it becomes difficult to deploy these frameworks in rural areas where there is no connectivity and no proper Internet connection. This paper focuses on energy-efficient and real-time detection of the severity of diabetic retinopathy on the low-powered edge device without any proper connectivity. In this paper, various deep transfer learning methods were investigated for DR detection, and these include ResNet50, Inceptionv3, EfficientNet-B5, EfficientNet-B6, and VGG19. These CNN models were trained on preprocessed APTOS dataset. To increase training data and to overcome overfitting, various data augmentation techniques were used. The highest accuracy of 86.03% was achieved by EfficientNet-B6.

Keywords Diabetic retinopathy · Transfer learning · Deep learning · Edge computing

Z. Maqsood (✉) · M. K. Gupta
Shri Mata Vaishno Devi University, Katra, J&K, India
e-mail: 19mms015@smvdu.ac.in

M. K. Gupta
e-mail: manoj.gupta@smvdu.ac.in

1 Introduction

Diabetic retinopathy (DR) is quite possibly the most well-known complexity of diabetes that affects the eyes and is the main source of visual impairment in a dynamic populace. The high blood sugar in the blood vessels of the retina causes damage to the eye retina [1]. Due to this, the blood vessels in the retina get blocked resulting in the leakage of fluids into the eye, and in the worst case, the retina tries to develop new blood vessels which do not develop fully and are immature, and the fluids like plasma leak easily into the eye and may cause severe vision loss. About 80% of individuals who had diabetes for a very long time or more create diabetic retinopathy. With the proper treatment and the regular examination of the eye, diabetic retinopathy can be detected at an early stage, and complete vision loss can be prevented [2]. The diabetic retinopathy is detected by the occurrence of different retina lesions in the eye image [3], and these include microaneurysms (MA)—red dot or balloon-like structures appear on the blood vessels, hemorrhages (HM)—red dots and represent actual bleeding, soft-cotton wool-like spots or white spots and hard exudates (EX)—small white or yellowish deposits. Based on the appearance of these lesions in the retina images, diabetic retinopathy (DR) can be classified into five stages [4] including (I) Stage 1: no DR, (II) Stage 2: mild, (III) Stage 3: moderate, (IV) Stage 4: non(pre) proliferative DR, and (V) Stage 5: proliferative. Table 1 summarizes the severity of the disease based on lesion findings in the funduscopy image. The various stages are depicted in Fig. 1.

In India, almost 80 million people are suffering from diabetes and comprise almost 17% of total diabetic patients in the world. With 9000–10,000 ophthalmologists estimated in India, the ratio becomes 1 ophthalmologist for the 100,000 population. This shows the serious shortage of ophthalmologists. Also, the manual detection of DR is a time-consuming process and even difficult for the doctor to evaluate/examine the fundus image and may lead to misdiagnosis. Also, to have real-time responses for some critical healthcare applications like detection of DR among masses and IoT applications, various researches have entailed edge computing and deep learning

Table 1 Stages of DR

Stage	Lesion findings/detection	DR severity
I	No abnormality	No DR
II	Only Microaneurysms	Mild DR
III	● Retinal dot or haemorrhages ● Microaneurysms ● Cotton wool spots	Moderate diabetic retinopathy
IV	Any of the accompanying ● At least 20 intra-retinal HM in every one of 4 quadrants ● Clear venous seeping in 2 quadrants ● Intra retinal microvascular anomaly	Non(pre)-proliferative diabetic retinopathy
V	● Neovascularization (abnormal growth of new blood vessels) ● Pre-retinal HM	Proliferative DR

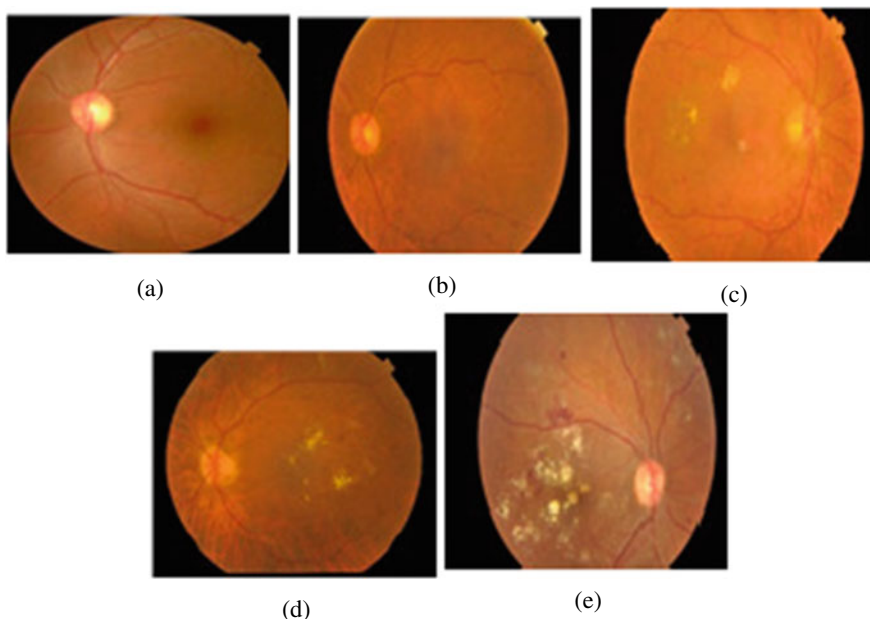


Fig. 1 Various DR stages: **a** No DR, **b**, Mild, **c** Moderate, **d** Pre-proliferative, **e** Proliferative DR

methods. In the references [6–15], they combined edge computing and deep learning for real-time responses where they leverage the computation task from the cloud to the edge to have minimized latency and reduced network traffic. Since most of the automated DR detection methods are cloud-based, i.e., the fundus images capture using fundus cameras/smartphones need to be transmitted to some central location (cloud) for processing or classification. Using this edge-cloud architecture still, we need connectivity and proper Internet connection. In areas where there is no connectivity and no proper Internet connection, it is difficult to use this cloud-based system for diabetic retinopathy detection [16]. Currently, fundus cameras are easily available in the market. The fundus cameras can be connected to the AI-powered device and detecting DR on-device. The motivation of this paper is to automatically detect diabetic retinopathy on-device and do the inference in real time.

2 Related Work

The consequences of diabetic retinopathy can be prevented by detecting and treating it at an early stage. After detecting the disease, an examination/evaluation needs to be done regularly to know the progress of the disease [17]. This can be effectively done by automatic detection which is time and cost-effective rather than manual examination. Over the recent past, the applicative span of deep learning has encompassed

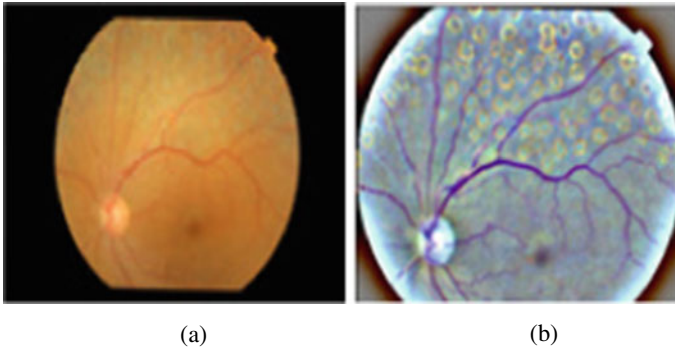
almost every field of human civilization like health care, manufacturing, space exploration, etc. In the field of medical image analysis, convolutional neural networks are widely used deep learning models for image analysis and are highly effective. To automatically detect diabetic retinopathy from the fundus images of the retina, different researchers have used different deep learning techniques. These techniques classify the fundus images into binary or multiple classes. Xu et al. [18] proposed a method that used CNN to automatically classify the 1000 images from the Kaggle dataset into diabetic retinopathy images or no DR images. Data augmentation techniques such as flipping, rotating, rescaling, translation, and shearing were used to increase the training data. The CNN method included eight CONV layers, four max-pooling layers, and two FC layers and achieved 94.5% classification accuracy. Esfahani et al. [19] used a pre-trained model ResNet34 and used images from the publicly available Kaggle dataset to classify them into no DR and DR images. Preprocessing of images like weighted addition image normalization and Gaussian filter was applied before feeding into the network. This method achieved 85% classification accuracy and 86% sensitivity. Pratt et al. [20] proposed a CNN method as a classification model. They used images from the publicly available Kaggle dataset and classified them into five different diabetic retinopathy stages. Preprocessing like resizing of images and normalization of color was done. They used the SoftMax function for the classification of 80,000 images. Ten convolutional layers, eight max-pooling layers, and three fully connected layers were included in their customized CNN architecture. The dropout method was used to overcome overfitting. This method achieved 95% specificity, 75% accuracy, and 30% sensitivity. The study studied by Wan et al. [21] used various pre-trained CNN models which include AlexNet, ResNet, GoogleNet, and VggNet coupled with hyperparameter tuning and do transfer learning to know how well these models classify. Before feeding the 35,126 images of the Kaggle dataset to the various models' image preprocessing like crop, normalize, and NLMP (denoising) was performed. To increase the training data, data augmentation was done. Among all the models, VggNet achieved maximum accuracy of 95.68, 97.86% of the area under the curve, and 97.43% specificity. Likewise, Khalifa [22] et al. also made use of transfer learning and studied six different pre-trained deep CNN models that include ResNet18, AlexNet, SqueezeNet, GoogleNet, VGG16, and VGG19. They used Asian Pacific Tele Ophthalmology Society (APTOS) dataset in their research. To overcome overfitting, different data augmentation techniques were used. The maximum testing accuracy of 97.9% was obtained by the AlexNet model.

3 Dataset and Pre-processing

The dataset used in this research to develop a model for automatic detection of diabetic retinopathy is Asian Pacific Tele Ophthalmology Society (APTOS) dataset. The APTOS dataset was published in the second quarter of 2019, containing 3663 fundus images, and the clinician has rated each image in five different classes (0–4) based on the severity of DR as given in table 2. The images were taken from different

Table 2 Different classes and no. of images

Class	Severity	No. of images in each class
0	Normal	1806
1	Mild DR	370
2	Moderate DR	999
3	Pre-proliferative DR	193
4	proliferative DR	295

**Fig. 2** Original (a) versus preprocessed image (b)

geographical locations and under different lighting conditions. Some of the images are overexposed, underexposed, or out of focus. Also, there is a high imbalance among the classes as the number of images in class 0 is much greater than in other classes.

We have rescaled all images to the same size as all the images were of different sizes. Also, some of the images were out of focus so, we have cropped each image by finding the center of the retina and crop to a square containing a circular area of fundus. For non-uniform illumination and blurring due to various lighting conditions, a Gaussian blur version of images was carried out to highlight the important spots. The original and the preprocessed images can be viewed in Fig. 2 below. Different data augmentation techniques were used to reduce overfitting and to increase training data to render models, and these include rotation, flipping (horizontally and vertically), shearing, and zoom-in.

4 Methods

Convolutional neural networks (CNNs) are the most common and widely used deep learning methods that have achieved great performance in the field of medical imagery [23]. Building a model from scratch to solve a certain problem is a

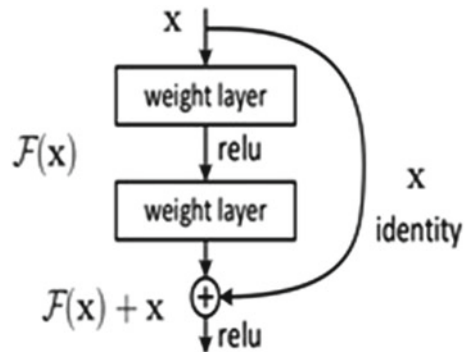
time-consuming and costly task. The model that has been trained to solve certain tasks can also be used to solve another related task [24]. This is what transfer learning is all about. Transfer learning is an approach, an optimization to save time and get better results. It has gained popularity in deep learning where pre-trained models that have learned from solving the problem with huge labeled training data and use this knowledge to solve some other related tasks. The last layers of the pre-trained models are deleted and viewed as feature extractors and are the starting point for a new related problem. Transfer learning allows to retrain the existing pre-trained model on the custom dataset due to which not only training time is reduced but also the dataset size. In our study, we have also used transfer learning and parameter tuning, we have used ResNet50, Inceptionv3, EfficientNet-B5, EfficientNet-B6, and VGG19, which are the latest deep convolutional neural networks, and do transfer learning to detect the severity of diabetic retinopathy.

4.1 ResNet 50

The subsequent architectures after the AlexNet that won the ImageNet competition in 2012 tried to add more layers in the deep neural network to reduce the error rate. Adding more layers to the network generally gives better performance up to some limitations. After the limitation, the performance degrades, and the problem associated with it is vanishing/exploding gradient.

To address this problem, Kaiming et al. [25] put forward a framework called residual network (ResNet) that won the ILSVRC championship (2015) that uses skip connections as shown in Fig. 3 or shortcut connections. If some block of the network hurts the performance, the block is skipped using these skip connections in the training process. ResNet50, a variant of the ResNet model, contains 48 layers of convolution, 1 max pool, and 1 average pool layer the total of 50 layers.

Fig. 3 Resnet architecture



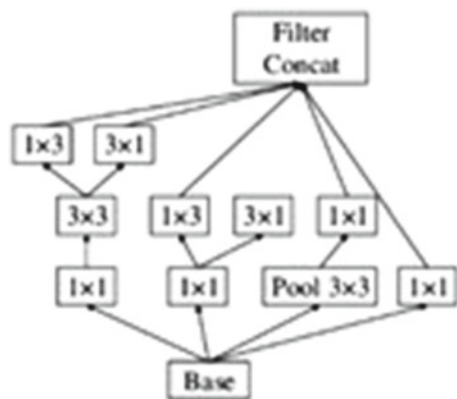
4.2 InceptionV3

Inceptionv3, a variant of inception [26] architecture, is a 48-layer deep convolutional neural network. It was introduced in 2014 by Google researchers and is trained on the ImageNet dataset that contains 1.2 million images approximately and can classify into 1000 different classes like keyboard, mouse, animals, etc. In previous versions of inception also, 1×1 convolution layers before 3×3 and 5×5 were used to reduce the parameters to enhance the utilization of computational resources. The inception network consists of various inception modules also called repeated modules are stacked one above the other to form the deep convolutional network. The Inceptionv3 module is shown in the figure below. For easier model adaptation in Inceptionv3, various techniques were suggested for optimizing the networks, and these include regularization, dimension reduction, factorized convolution, and parallelized computation.

4.3 EfficientNet B5 and B6

EfficientNet introduced by Tan and Le [27] in 2019 is a new technique achieving the state-of-the-art image classification accuracy from Google AI research. This technique rethinks the way we scale convolutional neural networks up. One way of scaling the CNN is to add more layers, for example, ResNet18 to ResNet200. Unlike in ResNet, systematic and compound scaling of depth, width, and resolution of convolutional neural networks is done in EfficientNet. The blocks of MobileNetV2 are used as core building blocks of EfficientNet and achieve better performance with low computing power. The family of EfficientNets ranges from B0 to B7. The base model of EfficientNet-B0 is similar to MnasNet whose architecture is given in Fig. 4.

Fig. 4 Inceptionv3 module



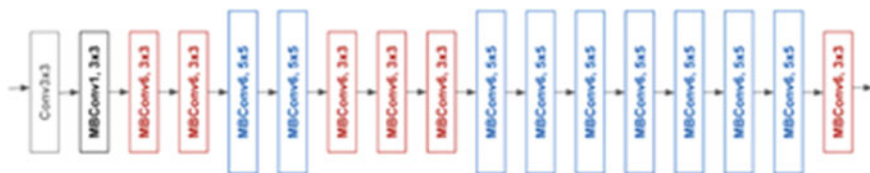


Fig. 5 EfficientNet-B0 baseline architecture

All the models of EfficientNet contain seven blocks. As we move from EfficientNet-B0 to EfficientNet-B7, the varying number of sub-blocks of these blocks increases (Fig. 5).

4.4 VGG19

In 2014, Andrew Zisserman and Karen Simonyan of Visual Geometry Group (VGG) lab proposed a convolutional neural network named VGG 16 [28] for the ImageNet competition and won first place. VGGNet consists of 16 convolutional layers with only $3 * 3$ kernels. Another variant of VGGNet is VGG19 which has 19 CONV layers. The core idea behind the VGG is to keep the CONV size small and constant and design a very deep network. The key difference between the VGG16 and VGG19 is number of layers.

5 Performance and Result

In our study, the performance of five convolutional neural networks, i.e., Resnet50, Inceptionv3, EfficientNet-B5, efficientNet-B6, and Vgg19 is evaluated to produce predictive models using cross-validation process.

The performance metric used to evaluate all these models here is accuracy and is given by

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

The average accuracy of these models to predict the severity of diabetic retinopathy is summarized in Table 3. The average accuracy of Resnet50, Inceptionv3, EfficientNet-B5, EfficientNet-B6, and Vgg19 are 82.66%, 79.48%, 84.38%, 86.03% and 82.59%, respectively.

Table 3 Classification accuracy of customized CNN models

Model	Resnet	Inceptionv3	EfficientNet-B6	EfficientNet-B5	VGG19
Training loss	0.0495	0.1684	0.1311	0.1291	0.3488
Training accuracy	83.01	0.9331	94.51	95.72	85.72
Validation loss	0.0564	0.8989	0.6094	0.5855	0.5138
Validation accuracy	82.66	0.7948	86.03	84.38	82.59

6 Deployment on the Edge

TensorFlow lite is an open-source framework that makes it easy to deploy the pre-trained machine learning models on the edge device. The deployment of models on the edge device like microcontrollers, embedded Linux devices, etc., can help improve latency, privacy, connectivity, and power consumption. TensorFlow lite converter (which converts and optimizes TensorFlow models to an efficient form) and TensorFlow lite interpreter (that runs the optimized models on different hardware's like microcontroller) are two main components of TensorFlow lite. The fundus camera to capture the fundus image of the retina can be connected to the low-powered AI device where the pre-trained model is already deployed to detect the severity of diabetic retinopathy in real time, energy efficient without proper connectivity. In our study, the customized EfficientNet-B6 has got highest classification accuracy, and therefore, we have converted this model to an optimized and efficient form using TensorFlow lite converter. To deploy this optimized model on the edge device and do the real-time inferences, we have taken Raspberry Pi which acts as an edge device and installed TensorFlow lite on it. The Raspberry Pi used here is 2B plus which is an old model having a 900MHz quad-core ARM cortex-A7 CPU and 1 GB of RAM. The optimized trained model is then deployed on the Raspberry Pi as a Web application. Here, we have used a PC/laptop that is connected to this edge device from where fundus image of the retina is taken and uploaded to Raspberry Pi for predicting the severity of diabetic retinopathy. The images from the PC are sent as post requests to the edge device through the Web browser. At the edge device (Raspberry Pi), the flask server installed on it takes the request and runs the script where inference takes place and the output class (severity) to which this image belongs is sent back to the browser. Hence energy efficient and in real-time severity of diabetic retinopathy is detected without any proper connectivity.

7 Conclusion and Future Scope

Diabetic retinopathy (DR) is quite possibly the well-known complication of diabetes that affects the eyes and is the main cause of blindness among active population. At the early stage, it may show no symptoms but in due course can cause blindness. The consequences of the DR can be prevented by detecting and treating it at an early stage. After the DR is detected, a regular examination is required to check the status of the disease. The manual detection of DR is time-consuming and costly and requires well-trained medical practitioners to evaluate the fundus images of the retina. Therefore, various automated methods have been developed for detecting DR, and most of these methods are cloud-based. In the areas where there is no connectivity and no proper Internet connection, it becomes difficult to make a diagnosis in such rural areas. In this paper, the attempt is to develop a deep learning model for detecting the severity of DR and deploy it as a Web application on an edge device. Therefore, the fundus images of the retina are classified into five stages based on the severity of the disease in real time without any transmission of data to the cloud. In this study, we have used CNN models including ResNet50, Inceptionv3, EfficientNet-B5, EfficientNet-B6, and VGG19 trained on the APTOS dataset and do transfer learning. EfficientNet-B6 has got the highest accuracy. To detect the DR on device, we deployed the efficient and optimized pre-trained EfficientNet-B6 on the Raspberry Pi which acts as an edge device and had a negligible impact on the classification accuracy.

References

1. Taylor R, Batey D (eds) (2012) Handbook of retinal screening in diabetes: diagnosis and management. Wiley, New York
2. Walter T et al (2002) A contribution of image processing to the diagnosis of diabetic retinopathy-detection of exudates in color fundus images of the human retina. *IEEE Trans Med Imag* 21(10):1236–1243
3. Agurto C et al (2010) Multiscale AM-FM methods for diabetic retinopathy lesion detection. *IEEE Trans Med Imag* 29(2):502–512
4. Kar SS, Maity SP (2017) Automatic detection of retinal lesions for screening of diabetic retinopathy. *IEEE Trans Biomed Eng* 65(3):608–618
5. Liang F et al (2020) Toward edge-based deep learning in industrial Internet of Things. *IEEE Internet Things J* 7(5):4329–4341
6. O'Donovan P et al (2019) A comparison of fog and cloud computing cyber-physical interfaces for Industry 4.0 real-time embedded machine learning engineering applications. *Comput Ind* 110:12–35
7. Ghosh AM, Grolinger K (2020) Edge-cloud computing for Internet of Things data analytics: embedding intelligence in the edge with deep learning. *IEEE Trans Ind Inform* 17(3):2191–2200
8. Rashid N et al (2020) HEAR: fog-enabled energy-aware online human eating activity recognition. *IEEE Internet of Things J* 8(2):860–868
9. Chang et al, Fog/edge computing for security, privacy, and applications
10. Zhang J et al (2018) Data security and privacy-preserving in edge computing paradigm: survey and open issues. *IEEE Access* 6:18209–18237

11. Xu X et al (2019) An edge computing-enabled computation offloading method with privacy preservation for internet of connected vehicles. *Future Gener Comput Syst* 96:89–100
12. Liu C et al (2017) A new deep learning-based food recognition system for dietary assessment on an edge computing service infrastructure. *IEEE Trans Serv Comput* 11(2):249–261
13. Wang J et al (2020) Deep anomaly detection in expressway based on edge computing and deep learning. *J Ambient Intell Hum Comput* 1–13
14. Tuli S et al (2020) HealthFog: an ensemble deep learning based smart healthcare system for automatic diagnosis of heart diseases in integrated IoT and fog computing environments. *Future Gener Comput Syst* 104:187–200
15. Zia Uddin Md (2019) A wearable sensor-based activity prediction system to facilitate edge computing in smart healthcare system. *J Parallel Distrib Comput* 123:46–53
16. Mathew G, Sindhu Ramachandran S, Suchithra VS (2020) EdgeAI: diabetic retinopathy detection in intel architecture. In: 2020 IEEE/ITU international conference on artificial intelligence for good (AI4G). IEEE
17. Abramoff MD et al (2008) Evaluation of a system for automatic detection of diabetic retinopathy from color fundus photographs in a large population of patients with diabetes. *Diabetes Care* 31(2):193–198
18. Xu K, Feng D, Mi H (2017) Deep convolutional neural network-based early automated detection of diabetic retinopathy using fundus image. *Molecules* 22(12):2054
19. Esfahani MT, Ghaderi M, Kafiyeh R (2018) Classification of diabetic and normal fundus images using new deep learning method. *Leonardo Electron J Pract Technol* 17(32):233–248
20. Pratt H et al (2016) Convolutional neural networks for diabetic retinopathy. *Procedia Comput Sci* 90:200–205
21. Wan S, Liang Y, Zhang Y (2018) Deep convolutional neural networks for diabetic retinopathy detection by image classification. *Comput Electr Eng* 72:274–282
22. Khalifa NEM et al (2019) Deep transfer learning models for medical diabetic retinopathy detection. *Acta Inform Med* 27(5):327
23. Anwar SM et al (2018) Medical image analysis using convolutional neural networks: a review. *J Med Syst* 42(11):1–13
24. Torrey L, Shavlik J (2010) Transfer learning. In: *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*. IGI Glob 242–264
25. He K et al (2016) Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*
26. Szegedy C et al (2016) Rethinking the inception architecture for computer vision. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*
27. Tan M, Le Q (2019) Efficientnet: rethinking model scaling for convolutional neural networks. In: *International conference on machine learning*. PMLR
28. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556)

A Survey on IoT Security: Security Threats and Analysis of Botnet Attacks Over IoT and Avoidance



M. Vijayakumar and T. S. Shiny Angel

Abstract IoT is an emerging technology that provides humans very handy support in various aspects and applications. This technology faces various threats in various aspects. The proposed work will analyze the various levels of threats and combating the threats. Various levels of threats are identified to the IoT. Over twenty-five, different levels of threats are identified for the IoT in different aspects. As the IoT is an emerging technology, it has to overcome these hurdles. In this paper, a nitty dirty review of the security-related challenges and wellsprings of peril in IoT applications is presented. Within the wake of talking around the security issues, diverse emerging and existing developments focused on finishing, and also, mainly Botnets-based threats feature over IoT is been provided solution as it is most vulnerable comparing other threats. Combating features are recommended.

Keywords Threats · Botnet attacks · Bargaining · Negotiation

1 Introduction

All recent technologies are having some sort of issues that makes the system handle with some sort of fear of safety; similarly, the IoT-based devices have the same issues. Most of the IoT devices are targeted because of certainly valid reasons as embedded components are easy to exploit, these devices are always in on condition, they follow low-security standards, even all users can be able to configure the device with a simple password that is easily accessible by the attackers, and developing malware can easily crack the password used as security in the IoT device. Monitoring and servicing of IoT are not well established for security. A single attack affects a large

M. Vijayakumar (✉) · T. S. Shiny Angel
SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India
e-mail: vm2893@srmist.edu.in

T. S. Shiny Angel
e-mail: shinyant@srmist.edu.in

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
D. P. Agrawal et al. (eds.), *Cyber Security, Privacy and Networking*, Lecture Notes in Networks and Systems 370, https://doi.org/10.1007/978-981-16-8664-1_13

141



Fig. 1 Different IoT structures

Table 1 Comparison of security between IT devices and IoT devices

IT security everywhere	IoT security
Devices with a lot of resources are common in IT	IoT devices need to be carefully provisioned with security measures
Far reaching IT is built on contraptions with a part of assets	IoT frameworks are composed of gadgets having confinements in terms of their program and equipment
For wide security and lower capabilities, complex calculation is actualized	As it was lightweight, algorithms are favored
Homogeneous innovation is mindful for tall security	IoT with heterogeneous innovation produces a huge sum of heterogeneous information expanding the assault surface

number of systems at a low cost, so attackers have an elation in attack on IoT devices. Figure 1 shows the past, present, and future architecture of IoT.

In the future, the contraptions (devices) are not fair anticipated to be related with the Web and other neighborhood contraptions (devices) but at the same time are required to talk with diverse contraptions (devices) on the Web authentically. Aside from the contraptions or things being associated, the thought of social IoT (SIoT) in addition creating. Social IoT will empower unmistakable social organizing clients to be related with the contraptions, and clients can share the contraptions over the Net [1].

With this colossal extend of IoT applications comes the issue of security and assurance. Without a trusted and interoperable IoT environment, rising IoT applications cannot arrive at ubiquity and may lose all their idle capacity. Nearby the security issues gone up against for the foremost portion by the Web, cell organizations, and WSNs, IoT also has its uncommon security challenges, for example, protection issues, confirmation issues, board issues, data stockpiling, etc.

Table 1 sums up different factors because of which making sure about IoT climate is substantially more testing than making sure about typical data innovation (IT) gadget (devices) so, in the proposed research work, the various vulnerabilities are

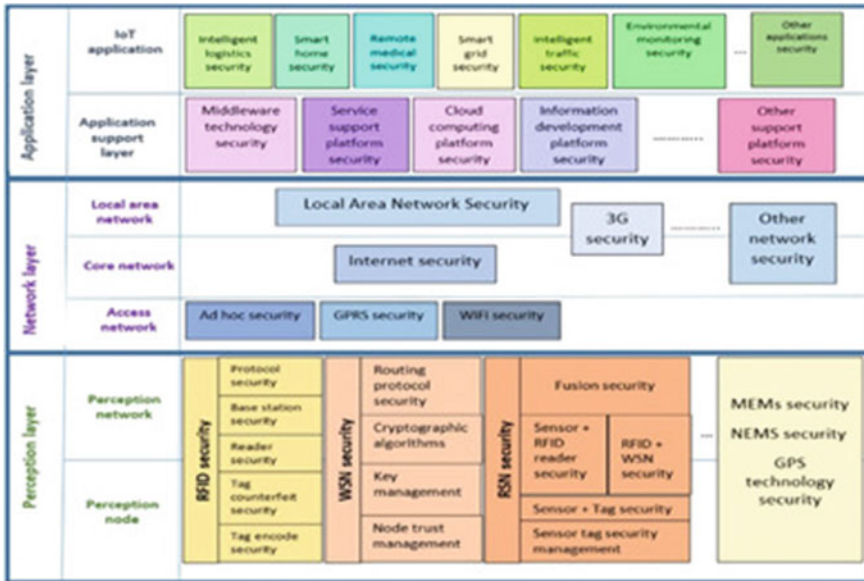


Fig. 2 Typical IoT architecture

discussed. Solution for those threats has been researched out as a research paper [1, 2].

1.1 IoT Security Architecture

Due to the differing qualities of the gadgets and huge number of communication conventions in an IoT framework, conjointly different interfacing and services offered, it is not reasonable to actualize security moderation based on the conventional IT organize arrangements. The current security measures which are connected in an ordinary organize may not be adequate. Assault vectors as recorded by the Open Web Application Security Venture (OWASP) concern the three layers of an IoT framework, which are equipment, communication interface, and interfaces/services. Thus, the usage of IoT security relief ought to envelop the security design at all IoT layers, as displayed in Fig. 2 [3].

There are different existing reviews on IoT security and protection issues. Yuchen et al. [4] have summed up different security issues in IoT applications. Hameed [5] was discussed many algorithms to secure the IoT network. In any case, these algorithms and strategies still need improvement in numerous angles to be utilized in the IoT framework and give confidence in the security and privacy environment. Ngu [6] focused mainly on the security issues related to IoT middleware and provides a detailed survey of related existing protocols and their security issues. Guizani et al.

[7] have reviewed different trust the executive's procedures for IoT alongside their advantages and disadvantages. Security components for IoT security, for example, software-defined networking (SDN) and network function virtualization (NFV), are discussed.

The main contributions of this work are as follows:

1. An arrangement of different IoT applications and unequivocal security and security issues were recognized with those applications.
2. A positive clarification of different threat sources in different layers of IoT.
3. Review on the proposed countermeasures to the security issues in IoT.
4. An examination of the open issues, troubles, and future examination headings for making secure IoT applications.

2 Sources of Security Threats in IoT Applications

As discussed in Section I, any IoT application can be divided into four layers: (1) sensing layer; (2) network layer; (3) middleware layer; and (4) application layer. Each of these layers in an IoT application uses diverse technologies that bring several issues and security threats. Figure 3 shows various technologies, devices, and applications at these four layers. This section discusses various possible security threats in IoT applications for these four layers [8].

2.1 Security Issues at Sensing/Physical Layer

Issue through Humans

There are many applications are being used on various devices by humans which are supported by the IoT device. Reference [9] Humans communicate with different devices in different forms for example in text mode, voice mode, and face-to-face mode. With single connectivity called the internet without the cybersecurity knowledge, people are not aware about the attackers in the digital world. People use most of the Internet-connected devices without the knowledge they are exposed to threats to their data, which would become a threat to their life itself in some cases. Most unsecured components could be a backend information provider of some organization or a corporate network. This should be secured with a certain research methodology that will provide complete protection.

Deficiency Technology Update

Most people lack in investing to have IoT-based infrastructures that will pave way for the attackers to make an easy entry for attacking the devices. The lack of proper update of installed devices also makes a chance of attack, and it makes device open for all to take up the data. Making proper updates will avert the breach of the data that will be major security assert for the concern [10].



Fig. 3 Layers in IoT System

Poor Physical Security

Software-based security is being discussed by most of the researchers, but there are chances of hardware-based security too, and the intruders and hackers planning access the device with the hardware devices too. The hardware protection should be enabled in a way that tampering of the device can be possible by the hackers, in that USB ports are one of the devices which could be able to tamper [11]. A seal can be proposed to the USB ports that would have an anti-tampering shield that would protect the device from the hacker. The shield will be embedded with the main circuit of the device, which would collapse the entire system if the device is meant to tamper, by that the hackers will not attempt to disassemble the device.

RFID Skimming

The hackers use this mode of attack to gain information about a transaction made through all transaction cards. The card detectors are compromised with RFID Skimming techniques supported by near field communication (NFC) device [12]. The device which is placed by the hackers will make a copy of the transaction data and will transfer it to the hacker’s system. It will mean to sense plain data from the device and transfer it to the hacker’s server.

2.2 *Security Issues at Data Link Layer*

Appliance Phishing

Phishing of machines will be an identified concern in the forthcoming period of years. Attackers will try to enter illegally and try to access IoT components and will send fake signals to networks that will cause owners large damage. The operating system will be in attack; for instance, if there is a plant controlled by an IoT device, the fake signals will be giving different signals that make the machinery large damaged. The varying and wrong signals will make the system a large troublesomely [13]. This increases chances of appliance phishing issue. This could be rectified through strong encryption-based chipping security.

Much components, More Coercions

As many devices are connected with IoT, they much have chances of attack by both active attackers and passive attackers. A device meant for support is the data centers at the backend. The product which is in use can store the personal information about the user and has the chance to propagate to the remote server, so there are chances of a copy of the secret personal information of the user to known to the third person. So, secret gateway should be derived at the IoT device itself to hold the data transfer or data store of the user over the device or the server [14].

2.3 *Security Issues at Network Layer*

Hazardous Communication

Most of the IoT components will not encode the communications while transferring to the networked systems. It is considered the largest safety task for IoT out there. IoT using concerns want to make certain conversations among gadgets and cloud server-based amenities in a steady and encrypted form. Great exercises to conform steady communicate are to practice delivery ciphered then to custom criteria similar TLS. Quarantining gadgets via the usage of diverse networks likewise facilitate produce stability in addition to a secluded conversation that maintains the communicated statistics stable as well as trusted [15].

Resident-Based Attack

Internet of Things (IoT) protection had become a freighting fear; subsequently, it links the opening among the digital and somatic sphere. By way of previously stated, unprotected Internet of Things (IoT) components may ooze user's connectivity (Internet Protocol) domicile that may not identify users living locality. Illegal control takers have chances to have a business by using collected data toward dissident Web services, place in which unlawful outfits. As well as, while the user in a make of Internet of Things (IoT) coupled wise household protection arrangements, after that this setup will have chances of negotiated too. For this reason, IoT device protection is stressed often. The user would like to protect his associated components via the Internet of

Things (IoT) protection as well as therefore by the support of tunnel-like networks as VPNs [16].

Individual Data Disclosures

Skilled computer-generated delinquents can be a reason for huge mutilation even though inspection data provides some of the network protocols (Internet Protocol) connectivity via unprotected IoT components. The connection modes often will not locate the utilizer's area as well as the user's original location they live. So, the virtual private network (VPN) is recommended by techies and experts. Setting up a tunnel network like VPN over the user router may encode the entire movement via ISP [17]. Virtual private network or tunnel networks have the user connecting privy Internet address as well as protect the user overall user resident connecting setup.

Privacy Concerns

Huge datum is being collected through most of the IoT devices that might include with most sensitive and secret information without any proper security aspect for the information. Users should review the security agreement made by the apps and the nature of the information being collected. If the information is more personal or sensitive, then the user to be cautious using that such apps [18]. Or a security application should be developed to encapsulate the sensitive and most private information while uploading apps. This encapsulation must be a security lock that could open by only the user. If it requires for the app service provider

2.4 Security Issues at Application Layer

IoT components drafted to Botnets

Alike other gadgets presence attacked in form of hackers who takes control of the device and email servers are changed into bulk junk mails or messages; clever device gadgets may also be customized in the form of vulnerable device code for carrying out distributed denial of service (DDoS) attacks. Earlier, attackers used infant displays-oriented output device watches to hold available huge gauge DDoS attacks. Producers want to recognize dangers linked with IoT-associated components and yield to vital actions to protect respective components. This attack is a danger, and it is to be countered with the aspect of security pattern, in an unbreakable server setup [19].

IoT device negotiation through junk Emails

The science and technology developments that occur date to date have made room for an overabundance of shrewd components into the usage of humans, but never stopped to shrewd utilizations, self-governing house control systems, etc. The components utilize the same computing energy by way of other IoT-linked components that have been utilized for many jobs. As per a new update, it has been identified that the devices that are been in negotiable condition can produce a huge amount of spam mails to perplex the user. For this issue, the server should be properly secured to counter this issue [20].

Resident-Based Attack

Internet of Things (IoT) protection had become a freighting fear; subsequently it links the opening among the digital and somatic sphere. By way of previously stated, unprotected Internet of Things (IoT) components may ooze user's connectivity (Internet Protocol) domicile that may not identify users living locality. Illegal control takers have chances to have a business by using collected data toward dissident Web services, place in which unlawful outfits rig function [21]. As well as, while the user in the make of Internet of Things (IoT) coupled wise household protection arrangements, after that this setup will have chances of negotiated too [22].

Negotiating Medicinal Procedures

Medicinal equipment coupled with the IoT have somewhat large chances of attack by hackers, who can take control over these devices and make eavesdrop on the important and secure medical and personal data of some important persons in society and even common man medical reports are to be secured in this money minded world, those records can be sold for some purposes.

For medical data protection, a chip can be inbuilt in the device to analyze the patients. The memory chip presents analysis, and device stores the present status of the the patient's conditions to make double encryptions. These chips are compatible in nature. These memory chips are given to the patient to maintain secrecy with him or herself to maintain secrecy, while he meets with his physician, a onetime password is generated and delivered to both doctor and patient personal mobile number to make the access of the memory device and get the information about the particular patient and provide the treatment [23].

Man-in-the-Middle Attacks

Eavesdropping mode of attack is made in this attack, hackers try to intercept the communication between the communicating persons through IoT device which would be insecure in nature or a dangerous network, masquerade attack is made over the users, and the attack makes a major bad impact over the user's major and most important information [24]. A security-based provision is proposed for this issue, a new technique is enhanced by combining the block chaining the IoT data and transferring the block chained the enciphered data through a tunneled network [25]. This would be a better approach which would be a strong network of secure IoT communication.

3 Common Attacks on IoT Devices

Shortage of development

Most of the devices based on IoT techniques do not adapt any protection aspect in their devices from hackers and data-based threats. A recent analysis warns of this aspect, around 30 million devices all over the world are used without any proper data security in their devices, and this will lead to any network-oriented attack over the devices. Most of the devices lack security updating. Even though they have a security aspect, they never update after a particular level. The concerns provide security to a

certain level only. This level is also not enough to combat the threats and attacks of hackers [26]. This makes users exposed to the hackers without any defending device. This should be made overcome by external security support, which might defend the system even though there is a stop in the security update for the concerned device.

Distant Contact

Reports discharged through Web data of Central Intelligence Agency conveys the about the USA. CIA have been controlling the devices illegally into IoT gadgets as well as changing the direction of the image capturing device/mouthpieces by deprived of the information on the proprietors. Indeed, the likelihood that assailants will occupy the gadgets in use by any user as well as store the proprietors deprived of their insights alarming and made utilized through no one additionally by the Administration itself. Its reports highlighted monstrous vulnerabilities in the most recent programming [25], for example, Android and iOS, which implies hoodlums can likewise exploit these vulnerabilities and complete preposterous wrongdoings.

Information Larceny

Hackers are usually after information which includes, however, no longer restricted to, patron names, purchaser location information, bank card numerical, monetary particulars, then additional. Alike while an organization has compact Internet of Things (IoT) protection, still some special assault courses by hackers may take advantage. In that case, such kind of tool is hooked up toward a concern's network establishments, the person who takes control over other devices is able to get benefit get right of entry toward the computer connected establishments as well as cull entire valued information [27]. Then, this information will be shared to illicit users for a huge sum by the foretold beneficiaries through the procedure as briefed earlier.

Computational Intelligence coupled IoT

Computational intelligence is an emerging technology that could IoT in countering the threats over it. The data storage-based threats can be somewhat averted through artificial intelligence technology. If the IoT device possesses this AI support. The technology will be providing support to control the IoT device based on the task. Automation can be defined as a code as IoT codes. In some cases, the IoT codes can also be interrupted by the hackers or attackers to change the activities of IoT devices by just changing the code and make the device work harmfully to the user itself [28]. So, there is both safety and security issue while using the artificial intelligence supported IoT device. An alternate technique should be sorted out to avert this issue to have safe usage of the IoT devices.

4 Evolution of Botnet

Generally, there are large numbers of Botnets are developed in the cyber environments for affecting various net-based devices. A short analysis is made about these Botnets. Botnets are generally classified into two Botnets; they are traditional Botnets and IoT-based Botnets.

4.1 Traditional Botnets

This Botnet is not segregated as the specific task of attack, it is meant for the attack of overall computational devices like computers and servers, and they attack the device with malware and zombies, compromise the device to act like malware and zombies. Botnet owners can control devices, they make the device to have denial of services to the users, and other attacks like spam mail and information theft are other attacks by the traditional Botnets.

4.2 IoT-Based Botnets

IoT-based Botnets are the system that forms a cluster that negotiated Internet of Things (IoT) components in the form of all electronic system devices which are already compromised by the Botnets infected by the malware. Malware will permit the assailants to make dominate the device making the task as a conventional Botnet. This IoT Botnet will replicate its patch with the connecting devices to which it connects and makes the device bot-affected device.

4.3 Different Botnet Attacks

For understanding Botnet attack outcomes, some of the attacks are elaborated, and they are as follows

Linux. Airdra

This attack is identified in the year 2012 through the cybersecurity scientists at ATMA.ES. It is the first identified and registered attack where a large number of telnet-connected devices are affected due to this attack.

Bashlite

This attack came to light in the year 2014 a source code is published with multiple variants in the type of Bashlite in the different names gayfgt, qbot, lizkebab, and torlus. Over 1 lakh devices had been affected due to this attack.

Mirai

This attack was made in the year 2016, and this attack made a record-breaking attack over the devices in the form of a DDoS attack on the devices like Krebs, OVH, and Dyn. The main aim of this Botnet is all electronic devices that support IoT, and featuring ten predefined attacks, the Botnet made down many server infrastructures and cloud service providers. Assaults are GRE floods and water torture attacks.

Linux/IRCTelnet

This attack was made in the year 2016 through the malware Must Die, The Internet of Things (IoT) Botnet is aiming at all electronic devices like (routers, DVRs, and

IP cameras). UDP and TCP flooding of data signal along with the IPV4 and IPV6 protocols support is the outcome of this attack.

4.4 IoT Botnet Monitoring System (IBMS)

This system is proposed to monitor the attempt of Botnet-based attacks over IoT devices to provide alerts in means of the nonce to the server in which other IoT devices are connected. This alert will be based on the time interval basis that will make the IoT suspend the communication among the devices connected with. Detection of attack is identified through the behavior of each network that is connected with the main server for the particular application.

The devices which attempt to attack or attempt to fire the Botnet initially will have the behavior change, the sequence signal from the device varies, the time interval of normal time signal and attack planned signal varies, and this one aspect is considered as the behavior based on the probability. Based on it, a training set is made to identify the affected Botnet node or malleolus node which attempts the node [29].

In other modes, an artificial intelligence approach can be handled to identify the devices which are meant for the attack of Botnets and going to become a Botnet [30, 31].

4.5 Bargaining and Negotiation Methodology for Botnet Identification

Bargaining is a communication technique between two people generally to accept one person's ideology by others. Similarly, in a multi-agent concept, two agents had given a task to solve it, and they communicate with each other. One agent generally makes another agent accept its action and the other agent to do the task given by the agent. The same concept can be applied to identify the Botnet-affected node or device and isolate the device from communicating.

Three types of signals are made to arise among the two devices, the commands are as follows,

Signal α —To accept the task

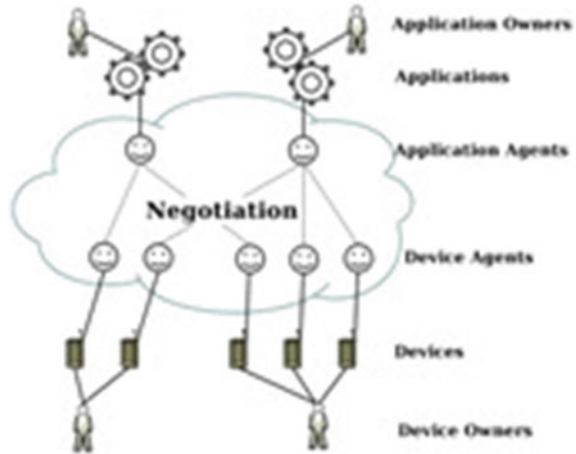
Signal β —The device is busy wait for n seconds

Signal γ —Negating the signal cannot accept the signal or task.

For signal transferring, three different wavelength signals can deploy for each different nonce modulated signal that can only be sensed by the specific sensors that are to be derived [11–13] (Fig. 4).

Step 1: The device starts to establish the communication with the connected device ($D1\alpha \Rightarrow D2$)

Fig. 4 Bargaining and negotiation among the agent in general



Step 2: Waits for t seconds for the response or to accept the task $D1\alpha(t)D2$

Step 3: Produces a Nonce α signal for the alert and negotiating signal to accept the task. (The negotiating signal nonce is made a maximum of three times to ensure the device is ready to take the task in a very short interval of time) $D1\alpha(n1)$, $D1\alpha(n2)$, $D1\alpha(n3).D2$

Step 4: Within the 3 nonce signal connected device will respond with the same $D2\alpha1$ nonce signal for accepting the task. $D2$

Step 5: If the signal $D2\alpha 1$ is received, then the device starts to give the command, for an application that is going to be processed. $D2$

Step 6: If the device is in the other task, it alerts with nonce $D2\beta$ instead of $D2\alpha 1 D2$

Step 7: The device stops sending nonce to that device and checks with other devices. $D2$

Step 8: If the particular device is affected by the Botnet of some issue nonce γ is arising from the devices. $D2$

Step 9: After receiving the signal $D\gamma$ the device disconnects with the issued device and stops communicating. $D2$

This mode of approach is meant to identify the devices that are not negotiating, and generally, most of the devices tend to negotiate to respond positively, if it or not attacked by the Botnet. This will be a better approach to identify the nature of the device. Whether it is in the position of executing the task or it is affected by any of the issues similar to any attacks [14–16].

5 Conclusion and Future Enhancement

In this paper, various IoT-based issues are identified; a suitable rectification is being provided for the issues. Solutions for Botnet attacks are been derived with a methodology that will identify and segregate the attacked device from other devices, which will stop the further breakdowns of the systems. In future enhancements, modulated signals are derived, with the modulated device sensing sensors, which will support identify the components which are been attacked, and this will be able to protect other components from further attack.

References

1. Frustaci M, Pace P, Aloï G, Fortino G (2018) Evaluating critical security issues of the IoT world: present and future challenges. *IEEE Internet Things J* 5(4):2483–2495
2. Hassija V, Chamola V, Saxena V, Jain D, Goyal P, Sikdar B. A survey on IoT security: application areas, security threats, and solution architectures. *IEEE Access*. <https://doi.org/10.1109/ACCESS.2019.2924045>
3. Sengupta J, Ruj S, Bit SD (2019) Comprehensive survey on attacks, security issues and blockchain solutions for IoT and IIoT. *J Netw Comput Appl* 149:102481
4. Yang Y, Wu L, Yin G, Li L, Zhao H (2017) A survey on security and privacy issues in internet-of-things. *IEEE Internet Things J* 4(5):1250–1258
5. Hameed A, Allometry A (2019) Security issues in IoT: a survey. In: 2019 International conference on innovation and intelligence for informatics, computing, and technologies (3ICT)
6. Ngu AH, Gutierrez M, Metsis V, Nepal S, Sheng QZ, IoT middleware: a survey on issues and enabling technologies. *IEEE Internet of*
7. Din IU, Guizani M, Kim B-S, Hassan S, Khan MK (2019) Trust management techniques for the internet of things: a survey. *IEEE Access* 7:29763–29787
8. Babar S, Stango A, Prasad N, Sen J, Prasad R (2017) Proposed embedded security framework for Internet of Things (IoT). Center for TeleInfrastruktur, Aalborg University, Aalborg, Denmark 2Tata Consultancy Services, Kolkata, India {sdb, as,np}@es.aau.dk, jaydip.sen@tcs.com, prasad@es.aau.dk. *Things J* 4(1):1–20
9. A framework of intrusion detection system based on Bayesian network in IoT Jie Wang College of Information and Communication Engineering Harbin Engineering University Harbin. *Int J Performabil Eng*, Oct 2018. <https://doi.org/10.23940/ijpe.18.10.p4.22802288>.
10. Security threats/attacks via botnets and botnet detection & prevention techniques in computer networks: a review. Emerald Simkhada Elisha Shrestha Sujana Pandit, Upasana Sherchand Akalanka Mailewa Dissanayaka and IT St. Cloud State University
11. A model to incorporate automated negotiation in IoT Mohammad Irfan Bala, Mohammad Ahsan Chishti Department of Computer Science and Engineering National Institute of Technology, Srinagar, India mirfan508@gmail.com,ahsan@nitsri.ne
12. Faratin P, Sierra C, Jennings NR (1998) Negotiation decision functions for autonomous agents. *Robot Autonomous Syst* 24(3):159–182
13. Jennings NR, Faratin P, Lomuscio AR, Parsons S, Wooldridge MJ, Sierra C (2001) Automated negotiation: prospects, methods and challenges. *Group Decis Negotiat* 10(2):199–215
14. Perera C, Zaslavsky A, Christen P, Georgakopoulos D (2014) Sensing as a service model for smart cities supported by the internet of things. *Trans Emerg Telecommun Technol* 25(1):81–93
15. Kang K, Pang Z, Da Xu L, Ma L, Wang C (2014) An interactive trust model for application market of the internet of things. *IEEE Trans Ind Inform* 10(2):1516–1526

16. Dementyev A, Hodges S, Taylor S, Smith J (2013) Power consumption analysis of Bluetooth low energy, Zigbee and ant sensor nodes in a cyclic sleep scenario. In: Wireless Symposium (IWS), (2013) IEEE International. IEEE 2013, pp 1–4
17. Bellifemine F, Caire G, Greenwood D (2007) Developing multi-agent systems with JADE, ser. Wiley series in agent technology. Wiley. Available: <http://books.google.hr/books?id=ZLBQAAAAMAAJ>
18. Zheng X, Martin P, Brohman K, Da Xu L (2014) Cloud service negotiation in the internet of things environment: a mixed approach. *IEEE Trans Ind Inform* 10(2):1506–1515
19. Raz Lin, Sarit Kraus (2010) Designing automated agents capable of efficiently negotiating with people—overcoming the challenge. *Commun ACM* 53(1):78–88
20. Choi SPM, Liu J, Chan S-P (2001) A genetic agent-based negotiation system. *Comput Netw* 37(2):195–204
21. Mukun C (2010) Multi-agent automated negotiation as a service. In: 7th international conference on service systems and service management (ICSSSM), 2010, pp 1–6
22. Bevan C, Fraser DS (2015) Shaking hands and cooperation in tele-present human-robot negotiation. In: Proceedings of the tenth annual ACM/IEEE international conference on human-robot interaction, Mar 2015, Portland, Oregon, USA. <https://doi.org/10.1145/2696454.2696490>
23. Oliver JR (1996) A machine-learning approach to automated negotiation and prospects for electronic commerce. *J Manage Inf Syst* 13(3):83–112
24. D'Angelo G, Palmieri F (2021) GGA: a modified genetic algorithm with gradient-based local search for solving constrained optimization problems, *Inf Sci* 547:136–162. ISSN 0020-0255. <https://doi.org/10.1016/j.ins.2020.08.040>
25. D'Angelo G, Castiglione A, Palmieri F (2021) A cluster-based multidimensional approach for detecting attacks on connected vehicles. *IEEE Internet of Things J* 8(16):12518–12527. <https://doi.org/10.1109/JIOT.2020.3032935>
26. D'Angelo G, Palmieri F (2021) Network traffic classification using deep convolutional recurrent autoencoder neural networks for spatial-temporal features extraction. *J Netw Comput Appl* 173:102890. ISSN 1084-8045
27. 25 Most Common IoT Security Threats in an Increasingly Connected
28. 20 Surprising IoT Statistics You Don't Already Know
29. 134 Cybersecurity Statistics and Trends for 2021. <https://www.varonis.com/blog/cybersecurity-statistics/>
30. Cyberattacks on IOT devices surge 300% In 2019, 'Measured In Billions', Report Claims. <https://www.forbes.com/sites/zakdoffman/2019/09/14/dangerous-cyberattacks-on-iot-devices-up-300-in-2019-now-rampant-report-claims/#61f7892a5892>
31. Symantec Security Center. <https://www.symantec.com/security-center/threat-report>

A Coherent Approach to Analyze Sentiment of Cryptocurrency



Ayush Hans, Kunal Ravindra Mohadikar, and Ekansh

Abstract In this paper, we have tried to analyze the real-time Twitter data of some popular cryptocurrencies, like Bitcoin. Bitcoin has been by far the largest cryptocurrency in terms of market size. Its market capitalization currently sits at over 1 Trillion US dollars. Even though it has gained popularity during this decade, still the cryptocurrency has seen many significant price swings on both daily as well as long-term valuations. In recent times, the influence of social media platforms like Twitter can be seen on cryptocurrency as well. Twitter is being used as a news source for many users who need to buy or sell Bitcoin. Therefore, understanding the sentiment behind the tweets which have a direct impact on price direction can help the user to trade in cryptocurrency better. The real-time data extracted using the APIs can help the user get the tweet sentiment at the time of purchase which can provide him/her an advantage in buying/selling the cryptocurrency. By combining the commonly used sentiment analysis tools like VADER and TextBlob into a single model, we can get a more accurate sentiment analysis of the tweets.

Keywords Sentiment analysis · Twitter · Tweets · VADER · TextBlob · Cryptocurrency · Blockchain

1 Introduction

As of April 2021, the market valuation of all the cryptocurrencies combined is near about \$2 Trillion US\$ after it doubled in 2021 in lieu of its increasing institutional demands. The value of Bitcoin itself had reached its all-time high of \$63,558, which was an increase of more than double in 2021. Many people see cryptocurrency as an actual currency that can be used for daily purposes, while others think of it as a great investment opportunity. Now more than ever, people have started to invest in cryptocurrencies like Bitcoin keeping in mind the high risks involved. For instance,

A. Hans (✉) · K. R. Mohadikar · Ekansh
National Institute of Technology Kurukshetra, Kurukshetra, Haryana, India
e-mail: ayushhans2011@gmail.com

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
D. P. Agrawal et al. (eds.), *Cyber Security, Privacy and Networking*, Lecture Notes
in Networks and Systems 370, https://doi.org/10.1007/978-981-16-8664-1_14

155

the value of Bitcoin is so unstable that sometimes, in a single day, the market value of a single Bitcoin fluctuates for more than 10,000 USD.

This type of volatility in the price of the cryptocurrency makes it difficult for the general person who wants to use it as an everyday currency and also for the traders who wish to invest in it for even a short term. Researcher Ladislav Kiroufek found that Bitcoin is a unique asset in that its price behaves in ways similar to both a standard financial asset and a speculative one [1]. Given that cryptocurrency prices do not behave like traditional currencies the prices are extremely difficult to predict.

Given that these cryptocurrencies use blockchain as their major technology, they are most likely to stay in the market for a very long period of time. This means we have to figure out an ingenious way to predict the price fluctuation before it even happens. Moreover, nowadays, social media has become an enormous source of information for all the major companies to monitor public opinion. Microblogging services like Twitter, Reddit have become the best known and the most commonly used platforms. Furthermore, they have evolved to become significant sources for every type of information [2]. Twitter is considered the most popular microblogging service that allows users to share, deliver, and interpret real-time, short, and simple messages called tweets [3]. Therefore, Twitter is used for providing a plethora of data that are used in the fields of opinion mining and sentiment analysis.

In this paper, we propose an approach that could be a probable solution for predicting the most accurate sentiments behind the flurry of tweets on Twitter about the particular cryptocurrency which can in turn help in predicting the price fluctuation. This can be implemented with the use of Twitter APIs which help us extract and store the real-time Twitter data about the particular cryptocurrency by using the keywords like Bitcoin, Ethereum. These tweets are collected, stored, and then analyzed to get to know the sentiment behind them. This sentiment analysis was done by using valence aware dictionary for sentiment analysis (VADER) and Textblob. VADER is a model which is used for sentiment analysis of text data that is sensitive to both polarity and intensity of emotion. On the other hand, Textblob is a Python library that is mostly used for processing textual data.

The remainder of this paper is structured as follows: Sects. 2 and 3 provide the background and a brief description of the related works in this field, respectively. In Sect. 4, we describe the various data sources used in the paper. In Sect. 5, we present in detail the proposed method. In Sects. 6 and 7, we discuss the results, provide a conclusion, and propose recommendations for some future work (Fig. 1).

2 Background

2.1 *Cryptocurrency and Blockchain Technology*

In this paper, we have analyzed Twitter data related to cryptocurrency. More specifically, we have extracted data related to the most used cryptocurrency which is Bitcoin.

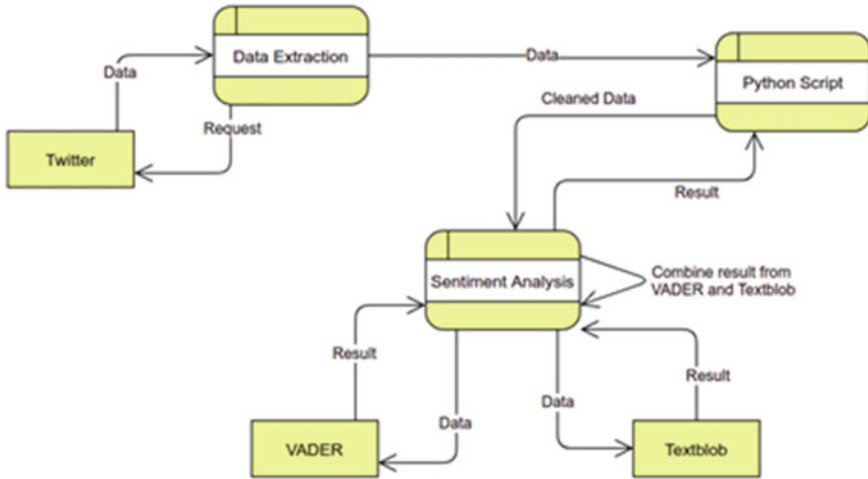


Fig. 1 Flow diagram of the model

Cryptocurrency is a digital currency that enables users to have an option for virtual payment. Bitcoin was developed by the pseudonymous Santoshi Nakamoto in January 2009 [4]. Santoshi Nakamoto also published a paper called “Bitcoin” along with the release of the cryptocurrency [5]. Cryptocurrency is based on blockchain technology. Blockchain is a digital ledger (digital book of financial accounts) that is used for recording information without any chances of hacking or changes in data. Key elements of blockchain technology are that it is decentralized, transparent, immutable, opensource, and the identity of the user who performs any kind of transaction is hidden or anonymous [6]. The data is stored not only at one location but at several locations. This technology ensures that people who are using the technology for digital currencies such as Bitcoin are the ones who are maintaining it. This also ensures that the cryptocurrency cannot be hacked, faked, or double-spent. The applications of blockchain go beyond cryptocurrency. Blockchain has various applications in the healthcare sector. Some examples of blockchain are in electronic health records and clinical research [6]. The data of clinical research or patients could be stored on blockchains and the records of individual users can be managed by themselves or just the user or organization authorized to manage it. This was proposed by Yue et al. [6]. This approach allows data to be managed safely and be distributed among different health organizations.

2.2 *Twitter*

Twitter is a microblogging service that enables users to share content through small posts. These posts are known as tweets. This social media platform enables users

who are not registered to also read the tweets. It was founded on March 21, 2006 [7]. The purpose of Twitter is to connect people. This is done when a user follows certain people who they think are interesting or find useful information from. This allows users to receive tweets of all the people they are following. Twitter is used widely to spread information on various trending topics, one of them being cryptocurrency, Bitcoin. There are numerous tweets every day on Bitcoin such as price changes, opinions, and facts. Twitter is also used for other purposes like sharing political news, promoting research, reaching out to people, retweets, receiving feedback on various topics or some product you have been developing, etc. The use cases of Twitter are countless. Users can use hashtags to post and the same topic separately [7]. A retweet is one of the powerful tools of Twitter which easily lets you post the tweet again to share some valuable information on that post quickly. It is one of the reasons why information spreads easily on Twitter.

2.3 Sentiment Analysis

Sentiment analysis is the mining of data through various tools such as natural language processing tools, text analysis tools to find out the sentiment of a particular text. Sentiment analysis can be done on various data from different sources such as Twitter, Google trends, and any other source which can provide a large dataset of texts or social media posts. Here, the tools are used to look for various words and emoticons in the text which when analyzed can tell us the sentiment of the text, whether it was in favor, neutral, or against a topic. Here, we are doing sentiment analysis on Bitcoin. The analysis is done using two tools, valence aware dictionary for sentiment reasoning (VADER) and Textblob. VADER is a rule-based model which is used for general sentiment analysis, especially for social media content [8]. This is based on semantic lexicons [8]. These are words that are given weight of positive and negative, and based on the collective score of these lexicons, we get a compound score. Words like good, great, and love are interpreted as positive sentiment, and words like hate, bad, ugly, and sad are interpreted as negative sentiment. Textblob is a Python library used to analyze textual data [9]. Unlike Vader is best used for purely textual data. It provides subjectivity and objectivity for each text, and using these parameters, we can determine the sentiment of the text.

3 Related Works

We have built this paper by doing wide research on topics related to sentimental analysis and price prediction of cryptocurrencies including but not limited to Bitcoin. A research paper by JS Lerner in the field of “Emotion and Decision Making” suggests the importance of sentiments in any decision [10]. This is also applicable in the field of cryptocurrency price prediction, which further strengthens the importance of

sentimental analysis in cryptocurrency price prediction. Toni Pano and Rasha Kashef published a paper on the same suggesting the use of VADER, while some papers use Textblob for doing the same [11]. Both the papers list the advantage of the libraries used. VADER works best for emoticons tweets, while Textblob works best for textual tweets [12]. Also, many articles and papers suggest that approximately 20 percent of tweets contain emojis [13]. We have used these results to develop a new approach using both the libraries, and the result is combined using a weighted average concept. The research paper presented uses the findings of all the above papers to extract the sentiment of the people about Bitcoin using the Twitter API for the data and gives the percentage of positive, negative, and neutral sentiments as the end result leading to the more accurate extraction of the sentiments which further improves the accuracy of applications such as price prediction of Bitcoin.

4 Data

To improve the prediction of sentiments related to Bitcoin, we have used the real-time data of Twitter using Twitter API. We have used Twitter as our source for data as it is the largest social microblogging platform with people active from all the fields and has people from all backgrounds which makes it suitable for extracting sentiments. The data is extracted from Twitter API by filtering it for the Bitcoin so as to get all the tweets related to the Bitcoin. This data is further processed using libraries, and mathematical operations are also performed so as to get the accurate prediction of sentiments. The dataset of tweets can vary from time to time because of its dynamic and real-time nature.

5 Methods

5.1 Sentiment Analysis Using TextBlob and VADER

After extracting the data and cleaning the data, now the data was analyzed to predict if it would be appropriate to use the data as input for our project. Even after cleaning, we felt the need to check if these tweets even have some real sentiments behind them. If all the tweets are not objective in nature, then performing analysis on them provides very little information to our model (Fig. 2).

As we know, Twitter is a microblogging platform where not all tweets are posted by humans. This can be confirmed by a report taken from Twitter which estimated around 40 million of its actual users as bots who post tweets that are merely for advertisements or provide only facts. Actually, even beyond bots, many conversations on cryptocurrencies are very subjective and can be very neutral in nature. Taking an example of a tweet containing the current rates of Bitcoin in terms of USD is a mere

Fig. 2 Analyzing the extracted data

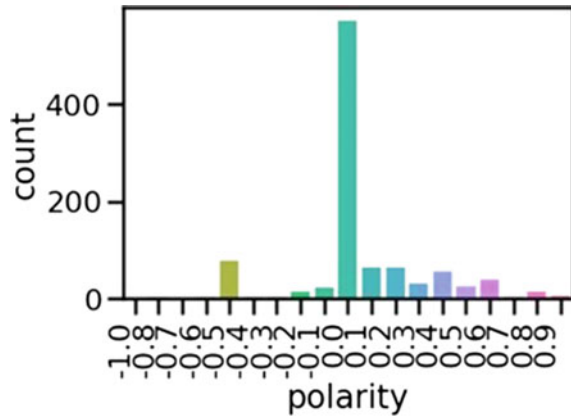
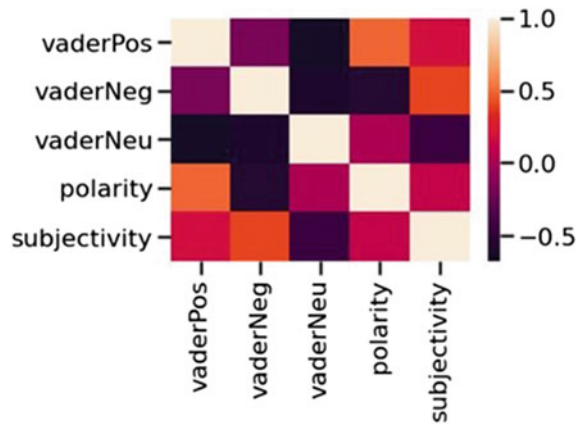


Fig. 3 Checking similarity between different parameters using HeatMap



fact and has no sentiments attached. Therefore, we need to segregate these tweets into positive, negative, and neutral and in this process keep in mind that most of these are words that are given weight of positive and negative, and based on the collective score of these lexicons, we get a compound score. Words like good, great, and love are interpreted as positive sentiment, and words like hate, bad, ugly, and sad are interpreted as negative sentiment. The tweets should be neutral due to the large presence of bots and many factual tweets (Fig. 3).

We use the VADER model to classify the cleaned data. While reading the tweets, the words are given weight of positive and negative, and based on the collective score of these lexicons, we get a compound score for that tweet. Words like good, great, and love are interpreted as positive sentiment, and words like hate, bad, ugly, sad are interpreted as negative sentiment. Thus, there are four fields of output after we provide the data to the VADER model. These are: Positive, Negative, Neutral, and Compound score. for j in tweet:

if(type(j) == type(string1)): tx = analyzer.polarity_scores(j) vaderPos.append(tx['pos']); vaderNeg.append(tx['neg']); vaderNeu.append(tx['neu']); vaderComp.append(tx['compound']) We also have used another Python library, TextBlob, for processing the textual data and classifying it. It provides subjectivity and objectivity for each tweet, and using these parameters, we can determine the sentiment of the tweet. Polarity lies between [-1, 1], where -1 denotes a negative sentiment, and 1 denotes a positive one. On the other hand, subjectivity lies between [0,1], and it quantifies the amount of personal opinion and factual information contained. The higher subjectivity means that text contains more personal information rather than factual information [14]. This score can intern be used to calculate the percentage of positive, negative, and neutral tweets from our data. This can be done by classifying all tweets having polarity above 0.4 as positive and all tweets having polarity below -0.4 as negative and the rest of the tweets as neutral. j in tweet: (type(j) == type(string1)): = tb(j) .append(.sentiment.polarity) .append(.sentiment.subjectivity)

5.2 *Incorporating the Output of both the VADER and TextBlob into One*

Usually, the tweets are not all textual and might contain some emoticons. This is often disadvantageous to process as it can handle only textual data with high efficiency. Also, VADER is used for both textual data emoticons, but its efficiency with textual data is not as good as TextBlob [15]. Therefore, if we were to combine the results of these two, then we would have a model which can analyze the sentiment better. Incorporating the scores of both VADER and TextBlob can be done taking into consideration that around 20% of the tweets posted on Twitter make use of emoticons. Taking a general guess, we can add the positive, negative, and neutral scores of both VADER and TextBlob into one by taking a weighted average of them. This indicates

$$0.2 * (\text{VADER_Positive}\%) + 0.8 * (\text{TextBlob_Positive}\%) = \text{Total_Positive}\%$$

$$0.2 * (\text{VADER_Negative}\%) + 0.8 * (\text{TextBlob_Negative}\%) = \text{Total_Negative}\%$$

$$0.2 * (\text{VADER_Neutral}\%) + 0.8 * (\text{TextBlob_Neutral}\%) = \text{Total_Neutral}\%$$

Thus, this can be used to get a better analysis of the sentiment of these tweets as this incorporated both the lexicon-based approach and the objectivity of the tweets to classify them. To prove our hypothesis, we picked up a training containing 2 million tweets and their sentiments and applied VADER, Textblob, and our custom model for sentiment analysis. The error percentage and efficiency of each of these three were monitored and to no surprise, it was as Table 1.

After getting positive results for our custom model on such a huge, we applied our model on real-time data which was extracted from Twitter API, and the results were as follows.

Table 1 Absolute Errors and Efficiency of the models

Models	Absolute error in %	Efficiency in %
TextBlob	9.28344	61.23
VADER	11.5172	49.41
Custom Model	4.459303	81.22

6 Results

After observing that our model works better, we applied it to the real-time data which was extracted using Twitter API and stored as a CSV file. We then cleaned the data and applied the classifying techniques to categorize the tweets as positive, negative, and neutral. In the case of Textblob, if the polarity is less than -0.4 , then the text is classified as negative. If it is above 0.4 , then the text is classified as positive. All other texts are classified as neutral. In the case of VADER, the classification of positive, negative, and neutral is based on the compound score. The compound score varies from -1 to 1 . If the score is less than -0.5 , then the text is classified as negative. If it is above 0.5 , then the text is classified as positive. All other texts are classified as neutral. As for our custom model, we took the weighted average of VADER scores and TextBlob scores. Research has shown that at least 20% of the tweets posted have emoticons and Textblob works better on Textual data while VADER works better with text containing emoticons. Using this information, we calculated the weighted average as shown in Fig. 4 and the adjoining pie chart (Fig. 5).

```

weightedAvgNegPer = 0.2*vaderNegPer + 0.8*txtNegPer
weightedAvgPosPer = 0.2*vaderPosPer + 0.8*txtPosPer
weightedAvgNeuPer = 0.2*vaderNeuPer + 0.8*txtNeuPer
print('weighted average Negative: ',weightedAvgNegPer,'%')
print('weighted average Posititve: ',weightedAvgPosPer,'%')
print('weighted average Neutral: ',weightedAvgNeuPer,'%')

```

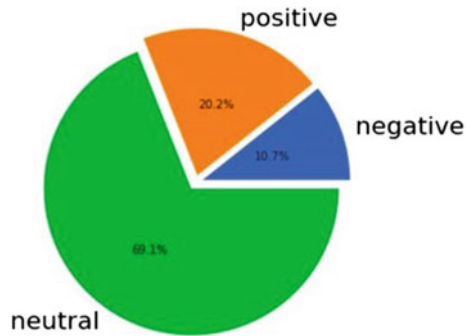
```

weighted average Negative: 10.72 %
weighted average Posititve: 20.220000000000006 %
weighted average Neutral: 69.06 %

```

Fig. 4 Calculating weighted average to form our custom model

Fig. 5 Pie chart representing the proportion of positive, negative, and neutral tweets



7 Conclusions and Future Plans

Earlier efforts to determine the sentiment related to cryptocurrency relied upon analyzing either textual data or emoticons optimally, i.e., one at a time. We tried to give weightage to both types of data so as to get the results as close as possible to the actual sentiment. We also tried to include subjectivity as a measure to determine the sentiment so that we can give more weight to the factual data rather than giving equal weightage to personal and factual data. In this way, we have tried to use several parameters using different tools and libraries to extract the sentiment. This work can further be used in price prediction models to predict the value of Bitcoin. Also, more complex models can be used to filter the noisy data to get more clean data which will further improve the accuracy of predicted sentiments. Also, data from other Internet sources can be incorporated with it to improve the data quality which in turn will improve the accuracy of our results.

Acknowledgements We would like to thank our college, National Institute of Technology, Kurukshetra, for giving us the platform to express ourselves. Also, we would like to thank our mentor Dr. B.B. Gupta, Asst. Professor, NIT Kurukshetra.

References

1. Kristoufek L (2015) What are the main drivers of the bitcoin price? Evidence from wavelet coherence analysis. PLOS ONE 10(4):1–15
2. Selvaperumal P, Suruliandi DA (2014) A short message classification algorithm for tweet classification. Int Conf Recent Trends Inf Technol 1–3
3. Singh T, Kumari M (2016) Role of text pre-processing in twitter sentiment analysis. Procedia Comput Sci 89:549–554
4. Farrell R (2015) An analysis of the cryptocurrency industry. Wharton Research Scholars (5-2015)
5. Abraham J, Higdon D, Nelson J, Ibarra J (2018) Cryptocurrency price prediction using tweet volumes and sentiment analysis. SMU Data Sci Rev 1(3), Article 1

6. Siyal AA, Junejo AZ, Zawish M, Ahmed K, Khalil A, Soursou G (2019) Applications of blockchain technology in medicine and healthcare: challenges and future perspectives. *Cryptography* 3(1)
7. Miraz, Maaruf A (2018) Applications of blockchain technology beyond cryptocurrency. *Ann Emerg Technol Comput (AETiC)* 2:1–6
8. Hutto C, Gilbert E (2014) VADER: a parsimonious rule-based model for sentiment analysis of social media text. *Proc Int AAAI Conf Web Soc Media* 8(1):216–225
9. TextBlob Homepage, <https://textblob.readthedocs.io/en/dev>. Last accessed 2021/04/21
10. Lerner JS (2014) Emotion and decision making. *Ann Rev Psychol* 66(1):799-823
11. Li TR, Chamrajnagar AS, Fong XR (2019) Sentiment-based prediction of alternative cryptocurrency price fluctuations using gradient boosting tree model. *Front Phys* 7(1):98
12. TowardsDataScience Page, <https://towardsdatascience.com/sentiment-analysis-vader-or-textblob-ff25514ac540>. Last accessed 2021/04/22
13. Emojipedia Page, <https://blog.emojipedia.org/emoji-use-in-the-new-normal/>. Last accessed 2021/04/22
14. TowardsDataScience Page, <https://towardsdatascience.com/my-absolute-go-to-for-sentiment-analysis-textblob-3ac3a11d>. Last accessed 2021/04/22
15. TowardsDataScience Page, <https://towardsdatascience.com/sentiment-analysis-vader-or-textblob-ff25514ac540#:~:text=Both%20libraries%20offer%20a%20host,%20with%20more%20formal%20language%20usage>

Supervised Machine Learning Algorithms Based on Classification for Detection of Distributed Denial of Service Attacks in SDN-Enabled Cloud Computing



Anupama Mishra and Neena Gupta

Abstract Software-defined network (SDN) is a networking technology that separates the data and control planes and enables centralized network control. This technique encapsulates lower-level functionality, allowing network managers to configure, manage, and regulate network behavior. While centralized monitoring is an important benefit of SDN, it can also be a serious security risk. The attacker gains access to the entire system if he successfully penetrates the central controller. Therefore, integration of SDN with the cloud itself provides insecurity to the cloud consumers. To skillfully implement DDoS over the controller, an attacker must gain access to multiple systems to launch multiple DDoS attacks. These DDoS attacks will deplete the controller's resources, causing its services to be unavailable. In order to detect controller attacks early on, it is critical to expand the coverage of the network. There are many existing techniques. However, relatively little research has been done in the area of SDN. A number of solutions fall under this category, including the use of machine learning algorithms for the task of classifying connections as either valid or invalid. To detect suspicious traffics and connections, we employ classification supervised machine learning algorithms, the Naive Bayes and support vector machine (SVM), which also achieved a promising result in order to verify the proposed work.

Keywords Machine learning · Cloud computing · SDN · DDoS

1 Introduction

Over the last few decades, traditional network architecture has remained largely unchanged and has proven to be cumbersome. SDN [1, 2], a new architecture, is well-suited to the dynamic and bandwidth-intensive applications. The architecture of SDN allows to control the network and functions for forwarding to be entirely separated,

A. Mishra · N. Gupta (✉)
Gurukul Kangri Vishwavidyalaya, Haridwar, India
e-mail: ngupta@gkv.ac.in

which facilitates direct programming of the network control. This separation breaks vertical integration as the control plane, which is what controls the forwarding of the traffic, is not integrated with the switches and routers that forward the traffic (the data plane). The control and data planes are separate, resulting in network switches being simple forwarding devices while the control logic is implemented in a logically centralized controller (or network operating system), reducing the effort needed to enforce policies and configure the network while also facilitating network evolution [3]. SDN controllers are often referred to as the SDN's brain and heart. If the controller is compromised, it is possible for the compromised user to take control of the entire SDN [4]. Distributed denial of service (DDoS) attacks are frequently used to target the controller. A DDoS attack is one in which several negotiated computer systems attack a target, such as a Web site, server, or other network resource, resulting in a denial of service (DoS) attack against the targeted resource. The flood of incoming messages, connection requests, or malformed packets causes the target machines to slow down or even crash and shut down, denying service to legitimate users or machines. The attacker deploys the necessary tools on the server via systems that have been compromised or negotiated. As with all computer systems, bots and zombies are at risk of DDoS attacks. For this reason, the research community has devoted much time and resources to mitigating these threats.

In this paper, we provide a technique for detecting DDoS attacks by applying support vector machines (SVM) and naïve Bayes classifiers. We use a training dataset to teach our classifier [5, 6]. The model is constructed using two classification techniques, support vector machines (SVM) and naïve Bayes. Once the server has received a new request from a client, it is sent to the model, which uses classification algorithms to predict whether the connection is normal or abnormal [7, 8]. Thus, by utilizing prior knowledge of connections and on top there is a classifier model, which can discover the traffic as normal or anomalous.

The rest of the paper are structured as follows: In Sect. 2, the related existing works discussed and analyzed. Section 3 proposed and discussed our approach, along with its fundamental concepts. Section 4 shows the implementation, the results and discussion are in Sect. 5, and finally, Sect. 6 ends the paper with conclusion.

2 Related Work

SDN is well-known for its low-cost functionality for network virtualization, dynamic policy enforcement, and centralized management over all network elements. The purpose of SDN is to replace existing networks by decoupling control logic from switches and routers. When the control plane is centralized, the network administrator bears a tremendous responsibility, as the administrator must also assure the network's security and good operation. The compromised network parts could be used to steal sensitive information such as personal information, for illicit reasons or to completely shut down the network. The architecture of SDN is layered, as seen in Fig. 1. An application layer, a control layer, and an infrastructure layer are the three

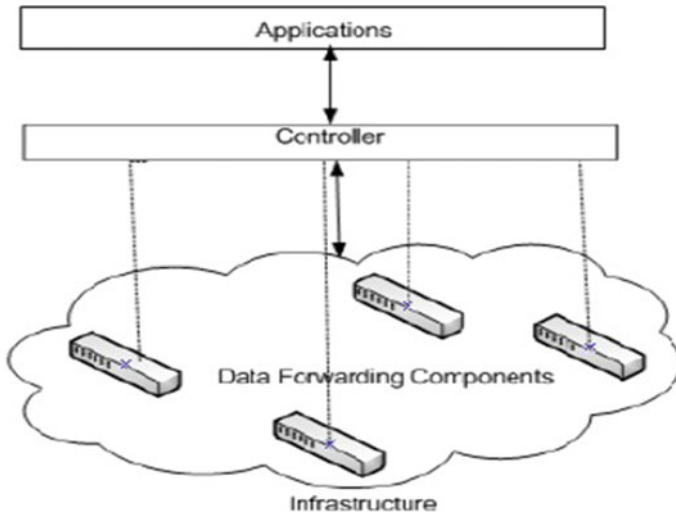


Fig. 1 Architecture of SDN

levels that make up the model [9].

- **Application Layer:** This layer provides user applications with both generic- and application-specific services. It is responsible for establishing, releasing, and terminating communication with other applications over a network.
- **Controller Layer:** The network's brain, the controller, is located in the control layer. The data required by the application layer is provided by it. The controller's information is used to construct SDN applications at the application layer. The SDN controller is responsible for converting application needs, coordinating network elements, and providing relevant information to SDN applications. A lot of applications compete for network resources with an SDN controller, all of which have different priorities.
- **Infrastructure Layer:** It consists of all network elements that allow network traffic to flow and make their capabilities available to the control layer via south-bound interfaces from the controller.
The south-bound interface, a control data plane, is the interface between the control system and the data system. At application layer, the applications over SDN-Cloud are developed, and they use north-bound interfaces to transfer to the controller plane their network requirements.

DDoS assault [10] is a well-known means of carrying out harmful activities by flooding a target system's resources with illegal traffic, causing the system to become

unable to meet user demands. The primary premise underlying DDoS attacks is that huge number of compromised nodes (zombies) are directed toward a single victim across numerous locations. The principal goal of attackers is to accomplish two things at once. The following is a list of them:

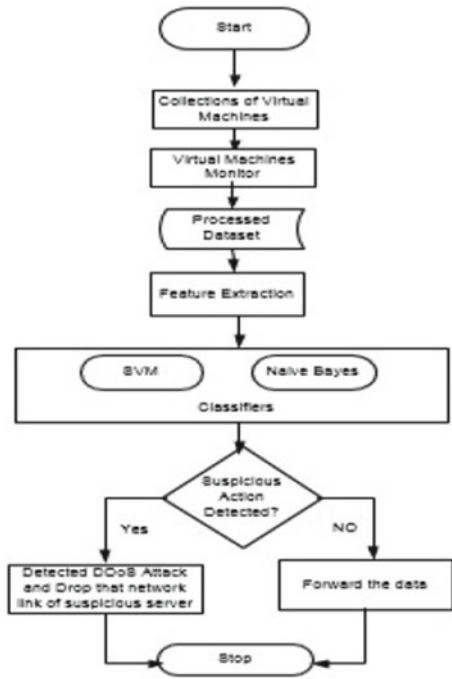
- (i) Running out of bandwidth.
- (ii) When all available resources have been exhausted [11]. Such assaults are particularly essential since they would have a negative impact on the server's performance and reputation. There are a lot of reasons why an attacker would carry out such attacks [2]. Several factors could be in play, including political gain, financial benefit, or simply disrupting service. SDN is unique in that it possesses a number of key qualities that make it possible to identify and defend against DDoS attacks. A central controller isolates the control plane from the data plane. The authors in [12] programmability that connects networks, traffic analysis that is facilitated by software, and automatic updating of forwarding rules are some of the distinctive aspects. SDN-based attacks are identified using a variety of methods, including entropy, connection rate, and machine learning [13].

1. Entropy-based: The entropy of a given attribute over a certain time period is a measure of its randomness. The randomness of a dataset has been effectively determined using a variety of entropy-based approaches [14]. A higher entropy number suggests a more distributed probability distribution, whereas a lower entropy number indicates a more concentrated distribution. As a result, these techniques are commonly employed for anomaly detection in classical intrusion detection systems.
2. Connection rate-based: There are two methods of anomaly detection based on connection rate. The first is the connection success rate, while the second is the total number of connections made. The former shows the percentage of successful connections compared to overall connections, while the latter shows the overall number of connections made over a given time period. A legitimate host has a better likelihood of connecting than a malicious host.
3. Traditional intrusion detection systems make extensive use of machine learning algorithms [15]. They have been used in both wired and wireless networks with great success. As a result, they have been demonstrated to be successful in detecting DDoS attacks in SDN. Based on particular network properties, these approaches identify connections as normal or anomalous.

3 Proposed Detection Method

In this section, we discuss the supervised machine learning algorithms based on classification techniques for detecting DDoS attacks in SDN. We implemented our proposed work using machine learning algorithms and the Ryu controller in Mininet

Fig. 2 Flow of proposed approach



emulator. Figure 2 presents flow of proposed approach which takes inputs for classifiers and based on algorithm decides whether the connection is normal or abnormal.

Classifier methods such as Naive Bayes and SVM networks were used to detect DDoS attacks.

3.1 Naive Bayes

A supervised machine learning algorithm called Naive Bayes [14, 16] is used to work with probabilistic models. In this model, probability is calculated for each class to determine their categorization, which is then used to forecast the values for a new class. x is an example of a problem that needs to be categorized. It can be represented as a vector by $x = x_1, x_2, \dots, x_n$ where n represents independent variables, and assigned to instance probabilities $p(cx/(x_1, x_2, \dots, x_n))$. There are K possible outcomes or classes for each of the ck possible outcomes or classes.

$$p(ck|x) = \frac{p(ck)p(X|ck)}{p(X)}$$

Table 1 Algorithm for detection of DDoS attack

```

Select and extract the features form virtual machine monitor
if the controller starts then
forwarded the extracted features to the classifiers
the number of connected hosts are counted
The counted value is classified by a classifier based on the probability
and it is appended in training dataset with value and its class label
end if
if the traffic is classified as class anomaly then
send alert messages and drop the connection
else
packet is forwarded
end if

```

3.2 Support Vector Machines

SVM's primary premise is to discover an ideal hyperplane that improves the dataset's generalization. It creates a model to predict whether or not a fresh sample will fall into one of the categories [17]. Let us say you have a training dataset $S = (x_1, y_1), \dots, (x_n, y_n)$. The transferred input vector is represented by x_i , and the target value is represented by y_i . SVM is a binary classifier with only two values for class labels: +1 or 1. SVM creates an ideal hyperplane H from the inputs that divides the data into various classes, and the hyperplane H can be defined as

$$x_i \in R^n : (\vec{w}, \vec{x}) + b = 0, \quad \vec{w} \in R^n, b \in R \quad (1)$$

The approach is based on applying the following function to identify the hyperplane that yields the largest distance between training samples.

$$f(\vec{x}) = \text{sign}(\vec{w}, \vec{x}) + b \quad (2)$$

Algorithm of detection of DDoS in SDN-enabled cloud is presented in Table 1. First it collects the features statistics from virtual machine monitors, and then, it will send it to network controller Ryu. It is forwarded to classifiers and based on the count value, and if is identified as anomaly, then the connection is dropped else the packet is forwarded.

4 Implementation

We use Mininet emulator to create topologies and assign hosts to the roles of server and client. Make client–server requests, and use them as test cases. As a result, the dataset is not used in its raw form for training; rather, it is used in a processed form. It has the number of parameters such as host time, no of requests, and the number of hosts connected in seconds. A secure channel enables the controller to switch to securely add and delete flow table entries. Whenever a new packet comes, the OpenFlow switch checks it in the flow table for a match. If there is no match in the flow table, the packet is forwarded to the controller for processing. The controller maintains an ACL, and MAC address of each packet is compared to the ACL. When a match is found, the packet is blocked, and connection is dropped; otherwise, the packet is forwarded to the server. If the proposed technique detects those hosts who are behaving abnormally, then they are denied access to the server, and also, their addresses are added to the ACL, in regards to assisting detection of their subsequent attacks.

5 Result and Discussion

The results of a simulation of a proposed algorithm in a Mininet environment are discussed in this section. Mininet is a network emulator that helps in creation and configuration of a network including hubs, switches, routers, host, and links. The switches in Mininet allow OpenFlow for custom routing. Figure 3 depicts the rate of accuracy for both the classification algorithm. The accuracy of a classifier is a measure of how successfully it divides positive examples into positive classes and negative instances into negative classes. The Bayesian algorithm can be as accurate as 63% of the time. It is possible that the classifier’s accuracy is due to the fact that it was fed a tiny dataset. The accuracy rate of SVM is substantially greater, at 82%. Fig. 4 shows the precision values for both the supervised learning methods. The number of occurrences classified as normal out of all the cases classed as normal is known as precision. The accuracy with which our algorithm classifies the cases as normal is represented by this value. The term “positive predictive value” is also used to describe it. The Naive Bayes algorithm has a precision of 76%. This means that it correctly classifies cases into the typical class 76% of the time. SVM has a modest advantage in terms of precision, with an 80% accuracy rate. SVM is clearly superior to Bayesian classification in case of normal class because it is more precise.

Figure 5 shows the recall values for both the classification-based supervised algorithms, which consider how many occurrences are correctly categorized out of all the others. The Bayesian method has a 60% recall rate. The recall of SVM is substantially higher than that of the Bayesian classifier, at 80%.

Fig. 3 Accuracy of Naive Bayes and SVM

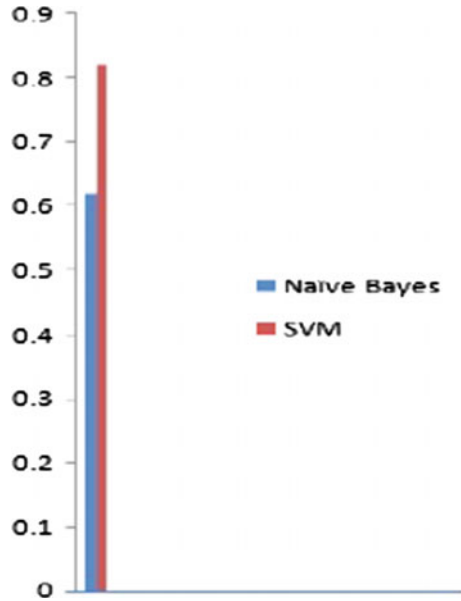


Fig. 4 Precision of Naive Bayes and SVM

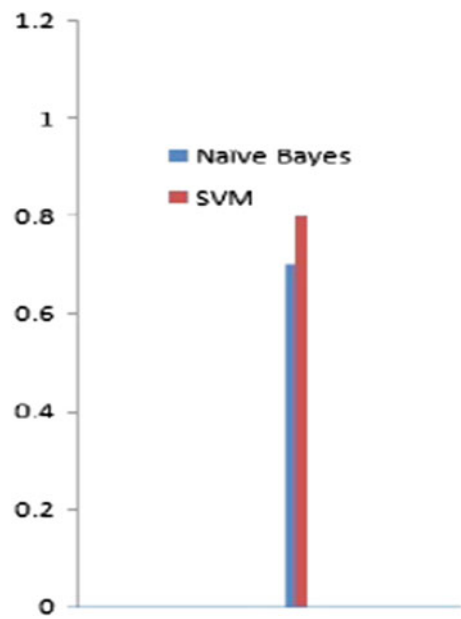
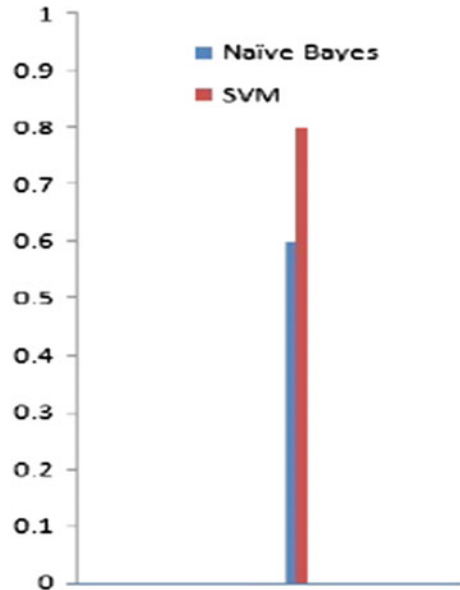


Fig. 5 Recall of Naive Bayes and SVM



6 Conclusion

Because it is not just dynamic but also manageable, cost-effective, and adaptive, software-defined networking is well-suited to the high-bandwidth, dynamic nature of today's applications. SDN promises to make it easier for network administrators to perform well to changing business needs. The security of SDN controllers is discussed in this paper. It presents how the supervised machine learning algorithms based on classification help in detection of DDoS. When the proposed work was implemented in SDN utilizing machine learning methods, it performed better. Mininet and the Ryu controller are used to accomplish the proposed methodology. By experimenting with various topologies, the findings indicate the solution's usefulness. Because of the two methods analyzed, the SVM method gives best results because it has the maximum accuracy and recall, as well as a fair precision value. In the future, the other machine learning algorithms like kNN, random forest, and linear regression can be applied.

References

1. Hussein A, Chadad L, Adalian N, Chehab A, Elhadj IH, Kayssi A (2020) Software-defined networking (SDN): the security review. *J Cyber Secur Technol* 4(1):1–66
2. Mishra A, Gupta N, Gupta BB (2020) Security threats and recent countermeasures in cloud computing. In: *Modern principles, practices, and algorithms for cloud security*. IGI Global, pp 145–161

3. Hou J, Fu P, Cao Z, Xu A (2018) Machine learning based DDos detection through NetFlow analysis. In: MILCOM 2018–2018 IEEE military communications conference (MILCOM), Los Angeles, pp 1–6
4. Rahman O, Quraishi MAG, Lung C (2019) DDoS attacks detection and mitigation in SDN using machine learning. IEEE world congress on services (SERVICES). Milan, Italy 2019, pp 184–189
5. Pise N, Kulkarni P (2016) Algorithm selection for classification problems. SAI computing conference (SAI). London 2016, pp 203–211
6. Al-Sharif ZA et al (2020) Live forensics of software attacks on cyberphysical systems. *Future Gener Comput Syst* 108:1217–1229
7. Zhang Z, Sun R, Zhao C, Wang J, Chang CK, Gupta BB (2017) CyVOD: a novel trinity multimedia social network scheme. *Multim Tools Appl* 76(18):18513–18529
8. Mishra A, Gupta N (2019) Analysis of cloud computing vulnerability against DDoS. In: 2019 International conference on innovative sustainable computational technologies (CISCT). IEEE, pp 1–6
9. Doshi R, Aporthe N, Feamster N (2018) Machine learning DDoS detection for consumer Internet of Things devices. IEEE security and privacy workshops (SPW). San Francisco, pp 29–35
10. Bhushan K, Gupta BB (2019) Distributed denial of service (DDoS) attack mitigation in software defined network (SDN)-based cloud computing environment. *J Ambient Intell Hum Comput* 10(5):1985–1997
11. Alsmirat MA et al (2019) Impact of digital fingerprint image quality on the fingerprint recognition accuracy. *Multim Tools Appl* 78(3):3649–3688
12. Al-Qerem A et al (2020) IoT transaction processing through cooperative concurrency control on fogcloud computing environment. *Soft Comput* 24(8):5695–5711
13. Gupta S, Gupta BB (2015) PHP-sensor: a prototype method to discover workflow violation and XSS vulnerabilities in PHP web applications. In: Proceedings of the 12th ACM international conference on computing frontiers, pp. 1–8
14. Das S, Mahfouz AM, Venugopal D, Shiva S (2019) DDoS intrusion detection through machine learning ensemble. In: 2019 IEEE 19th international conference on software quality, reliability and security companion (QRS-C), Sofia, Bulgaria, 2019, pp 471–477
15. Dahiya A, Gupta BB (2020) A reputation score policy and Bayesian game theory based incentivised mechanism for DDoS attacks mitigation and cyber defense. *Future Gener Comput Syst*
16. Dincalp U, Güzel MS, Sevine O, Bostanci E, Askerzade I (2018) Anomaly based distributed denial of service attack detection and prevention with machine learning. In: 2nd international symposium on multidisciplinary studies and innovative technologies (ISMSIT). Ankara 2018, pp 1–4
17. Khuphiran P, Leelaprute P, Uthayopas P, Ichikawa K, Watanakeesuntorn W (2018) Performance comparison of machine learning models for DDoS attacks detection. In: 22nd International computer science and engineering conference (ICSEC). Chiang Mai, Thailand 2018, pp 1–4
18. Dahiya A, Gupta BB (2020) Multi attribute auction based incentivized solution against DDoS attacks. *Comput Secur* 92:101763

Edge Computing-Based DDoS Attack Detection for Intelligent Transportation Systems



Akshat Gaurav, B. B. Gupta, and Kwok Tai Chui

Abstract Vehicular ad hoc networks (VANETs) are a critical component of intelligent transportation systems (ITS). Because VANET allows the transmission of critical and life-saving information between vehicle nodes, any effort to compromise the network should be recognized immediately, if at all feasible. The distributed denial-of-service (DDoS) assault is one kind of cyber-attack that affects VANET systems' availability. As a consequence of the DDoS assault, vehicle nodes are unable to transmit vital information. In this context, this experiment proposed edge computing-based DDoS detection techniques. The proposed technique uses packet entropy to distinguish DDoS attack traffic from normal communication. To determine the entropy values, we performed an in-depth study of five different machine learning methods. precision, accuracy, f1 measure, and recall are used to assess the performance of different machine learning methods

Keywords VANET · Entropy · Machine learning · Side channel attacks

1 Introduction

VANET [8, 10, 19] is a state-of-the-art technology that alerts vehicles to weather conditions and other municipal and traffic department operational components. ITS encompasses VANET, a subset of MANET [4, 9]. When the United States Fed-

A. Gaurav
Ronin Institute, Montclair, NJ 07043, USA

B. B. Gupta (✉)
Department of Computer Science and Information Engineering, Asia University, Taichung 413,
Taiwan
e-mail: gupta.brij@gmail.com

K. T. Chui
School of Science and Technology, Hong Kong Metropolitan University, Clear Water Bay, Hong
Kong, China
e-mail: jktchui@ouhk.edu.hk

eral Communications Commission designated a 75 MHz bandwidth of the 5.9 GHz band for Dedicated Short-Range Communication (DSRC) in 1999, it established the foundation for the establishment of VANET. When the ASTM standards committee selected IEEE 802.11a as the DSRC working protocol in 2001, VANET captured the interest of researchers. The IEEE updated the 802.11a protocol in 2004 and started development with the VANET standard wireless access in vehicular environments (WAVE).

The primary purposes of VANET are to guarantee passenger safety and to enhance the driver’s operating conditions. It is possible to save lives and resources by using VANET, because it can provide drivers with all the essential information, such as the location of medical centres and fuel stations, and information on traffic congestion. VANET attracts a large number of researchers because of its life-saving potential. The VANET is comprised of fixed roadside units (RSU) and movable vehicles equipped with onboard units (OBU) [13]. In the VANET, communication occurs between mobile vehicles (V2V) or between vehicles and RSUs (V2I). Through V2V communication, vehicles share crucial information with other drivers. The primary objective of this interaction is to provide more details to the vehicle’s driver about the surroundings. The important information for the driver of the vehicle, such as the nearest gas station or emergency, is transmitted through V2I connection between vehicles and RSU. Figure 1 depicts the fundamental VANET scenario, in which moving and parked vehicles communicate with one another as well as with RSU.

Due to the fact that VANET is based on wireless connectivity, it is susceptible to a range of cyber-attacks. A distributed denial-of-service (DDoS) attack is one of the cyber-attacks that is used by the attacker to gain control over the vulnerable vehicle. As a consequence of the DDoS attack, the vehicle node is unable to understand the information received from its neighbours or to send crucial information to its neighbours. Passengers’ safety is endangered as an outcome of the DDoS attack.

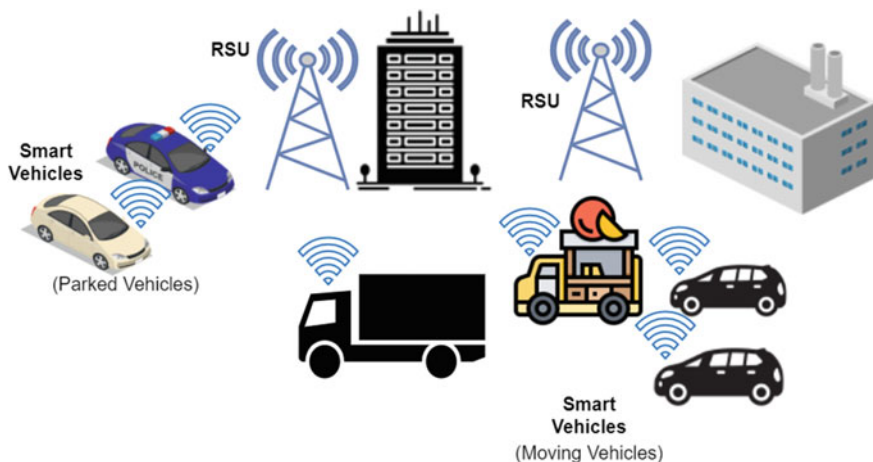


Fig. 1 VANET architecture

Many researchers proposed different DDoS detection methods [6, 7, 16, 18, 20, 21], but those detection techniques are not able to detect DDoS attacks efficiently. Fog/edge computing is one of the security solutions that is recently proposed by researchers. According to Cisco [15], fog computing is a decentralized computing concept that brings storage, processing, and other data centres closer to the end device. In fog/edge computing, fog/edge servers are located near to end devices to minimize end device latency. Each fog/edge server is analogous to a lightweight cloud server [11]; as a consequence of these features, fog/edge computing may be used to rapidly detect rogue nodes. Recently, VANET introduces the concept of fog/edge computing. In a VANET, fog/edge nodes analyse the network and stores important data in fog/edge servers. The fog servers, which are situated close to the edge devices, analyse the incoming data and make quick judgments, which enables them to detect the rogue car node. As a consequence, fog/edge devices are an easy method to increase security.

In this context, we developed a DDoS defence model that allows the use of machine learning to distinguish between DDoS attacks and regular traffic. We used entropy to differentiate DDoS attack traffic from normal traffic. Entropy is a measure of unpredictability, and as a result, it is an excellent technique for determining network traffic variance. To determine the most accurate approach for distinguishing between DDoS attack and normal traffic entropy values, we tested six different machine learning techniques: LR, SVM, RF, DTC, GBC, and MLA.

The remaining part of the paper is split into the following categories: Sect. 2 examines past research in the area of DDoS detection and mitigation. In Sect. 3, the proposed approach outlined is explored in detail. The study will then proceed to Sect. 4, in which the results of the implementation will be assessed. Section 5 presents some research challenges, and Sect. 6 of the chapter serves as the paper's conclusion.

2 Related Work

In this section, we review some of the works in the field of DDoS detection in VANET environment.

The authors in [12] propose a method for detecting malicious nodes in VANETs using fog. In this approach, the fog server gets information about the vehicle node and its topology from the cluster head. At the fog server, the reputation value of each node is calculated, and a malicious node is identified based on this reputation value.

The authors in [13] proposed a cluster head-controlled detection technique for VANETs. Under this architecture, the VANET network is divided into several clusters, each with its own cluster head. The cluster head produces a trace file for each node in the cluster. Each node's attack signature sample rate is calculated using the session's trace file, and a malicious node is detected using CCA.

The authors in [5] propose an efficient data downloading method in VANET by using a fog layer. The RSU identifies popular data based on vehicle node requests

and stores it at nearby edge devices, enabling any vehicle node to directly receive it. This increases the downloading efficiency of the vehicle node.

The authors in [8] point to a technique through which malicious traffic can be filtered out from the VANET environment. The proposed approach is implemented on RSU and uses an entropy-based approach.

The authors in [1] propose a hybrid approach based on machine learning for identifying rogue nodes in a VANET. The proposed method is a mix of the SVM kernel techniques used by AnovaDot and RBFDot. Collision packet drop jitter is utilized to train the algorithm and then detect the malicious vehicle node in the suggested approach.

The authors in [14] developed a technique for detecting DDoS attacks in a cloud environment that is based on SDN [3]. To distinguish DDoS attacks from regular traffic, the suggested method exploits the entropy shift. Statistical methods demonstrate that the suggested strategy has a high rate of detection, a low rate of false positives, and an outstanding capacity for mitigation. However, the proposed technique is only applicable in a cloud environment.

3 Proposed Mythology

We propose an entropy-based DDoS attack detection method for VANETs in this chapter. Our suggested approach is divided into two stages: the entropy calculation phase and the machine learning phase. We built a model to test our hypothesis that the attacker used bogus packets to flood the VANET network, and that the fraudulent packets were either generated by bots or by a DDoS attack tool.

3.1 Entropy Calculation Phase

In the proposed architecture, each vehicle node sends data to its one-hop neighbour until the data reaches the RSU. In our proposed approach, RSUs are considered edge nodes. A single edge node communicates with the edge nodes of several areas, enabling data sharing between them. As a consequence, if a malicious node moves across areas, our proposed method can also detect it. Figure 2 illustrates the topology of our proposed approach, with Edge node 1 and Edge node 2 connected to regions 1 and 2, respectively. Each edge node estimates the entropy of receiving packets for a specified time frame using Eq. 1.

$$H(Y) = - \sum_{i=1}^n P_i \times \log(P_i) \quad (1)$$

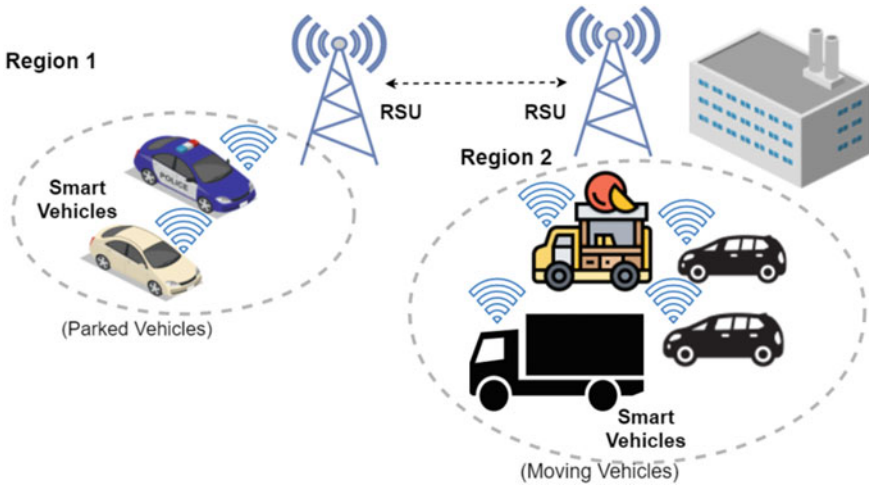


Fig. 2 System model of proposed approach

where $H(Y)$ denotes the Entropy of a randomly generated variable Y with n potential values, i.e. y_1, y_2, \dots, y_n and each value has a specific probability distribution P_1, P_2, \dots, P_n

3.2 Machine Learning Phase

In the preceding phase, we calculated the entropy of incoming traffic and prepared the dataset for future analysis and use in the next step. The purpose of this phase is to evaluate our dataset in order to determine which machine learning method is the most effective at distinguishing DDoS attack traffic from regular traffic. To examine the dataset, we utilized six of the most commonly used machine learning methods, which are listed below.

1. Support Vector Machine (SVM)—SVM is used to determine the best hyperplanes for classifying different points in an N -dimensional space. The hinge loss function (Eq. 2) is used in SVM to determine the largest possible margin between data points.

$$h(x, y, f(x)) = \begin{cases} 0, & \text{if } y * f(x) \geq 1 \\ 1 - y * f(x), & \text{else} \end{cases} \quad (2)$$

2. Logistic regression (LR)—LR used a liner equation (Eq. 3) to classify the data points. We used sigmoid function (Eq. 4) to limit the output (Eq. 5) from the linear equation.

$$\text{Linear equation}(z) = \theta_0 + \theta_1.i_1 + \theta_2.i_2 + \dots \quad (3)$$

$$\text{sigmoid function}(m) = \frac{1}{1 + e^m} \quad (4)$$

$$\text{Output}(y) = \frac{1}{1 + e^z} \quad (5)$$

3. **Decision Tree Classifier (DTC)**—Decision trees (DTs) are a quasi-supervised learning approach for classification and regression. The objective is to infer fundamental decision rules from data features in order to construct a model that forecasts the value of the dependent variable. A tree is a cost function constant's approximate value.
4. **Random Forest (RF)**—In classification and regression, random forests, also known as random choice forests, are a supervised learning method that works by training a large number of decision trees in a given problem domain. When used for classification tasks, the random forest's output is the class that has the greatest number of trees in it.
5. **Gradient Boosting (GB)**—Gradient boosting is a machine learning technique for regression, categorization, and other challenges that generates a prediction model from a collection of weak prediction models.
6. **Multinomial Naive Bayes (MNB)**—MNB is a technique for probabilistic learning that is frequently used in natural language processing. The Bayes theorem is used to forecast the tag in the independent variable. It determines the likelihood of each tag in a sample and returns the tag with the highest probability.

4 Results and Analysis

ONMET++ [22], Veins [17] and SUMO [2] were used to simulate the proposed approach, which we found to be quite accurate. SUMO is a road traffic simulator

Table 1 Simulation parameters

Term	Value
Simulation area	250 × 250 m ²
Simulation time	200 s
Routing protocol	Random
Traffic generation	Random
Normal traffic rate	1 packets/s
Attack traffic rate	5 packets/s
MAC layer	802.11p
Network interface	OMNET++
Network mobility framework	Veins
Traffic generator	SUMO

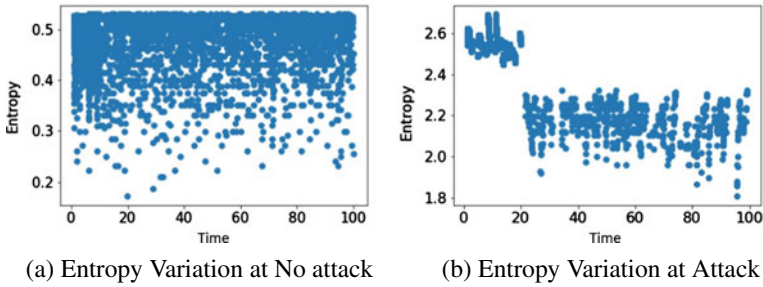


Fig. 3 Dataset representation

that may be used to simulate and analyse real-world traffic as well as other road management systems. Veins is a free, open-source simulator for modelling automobile networks that may be downloaded off the Internet. OMNET++ is a free and open-source event-based simulator that may be used to analyse and generate outcomes from a sequence of discrete occurrences. The simulation specifications are represented in Table 1.

4.1 Dataset Generation and Preprocessing

The simulation is carried out with the help of Veins, which links OMNET++ and SUMO. In the scenario, the attack packets try to overload the victim's vehicle with a large volume of erroneous traffic. In the case, the attacker node generates packets once every 1 s, while the legitimated nodes create packets once every five seconds. The whole simulation takes 100 s to complete, and all the log data is collected during the process. Since our proposed approach is not dependent on any particular routing protocol, we simulate it using a generic routing protocol. Null values are eliminated from the dataset during the preparation stage since the dataset includes a large number of null values. The variation of entropy during DDoS attack time and no attack time is represented in Fig. 3a, b. Finally, the dataset has been split into two parts: a training dataset and a testing dataset, allowing for a thorough assessment of the proposed method on both datasets.

4.2 Machine Learning Analysis

We utilized six different machine learning methods to analyse the dataset, and we calculated a confusion matrix in order to compute the false positive, true negative, false negative, and true positive values. With the assistance of the confusion matrix, we can quickly assess the effectiveness of machine learning methods in various

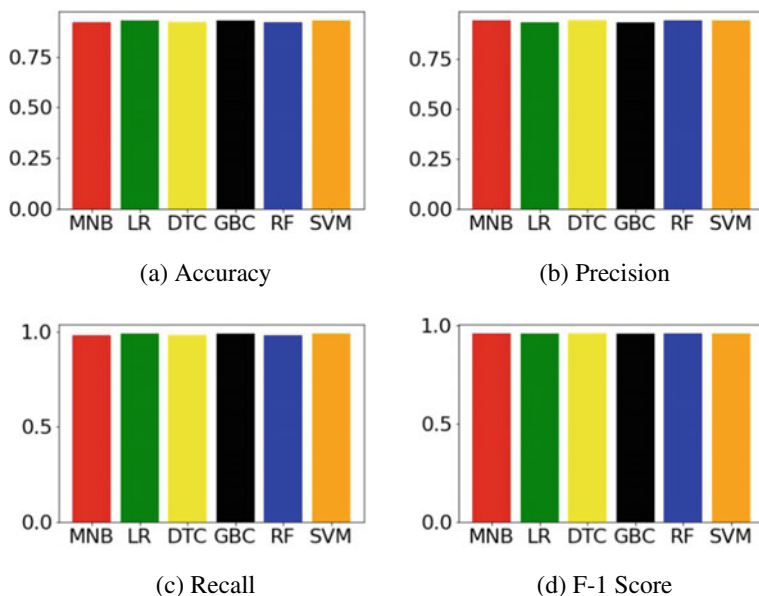


Fig. 4 Statistical parameters calculation

Table 2 Variation of statistical parameters of ML techniques

Parameter	MNB	LR	DTC	GBC	RF	SVM
Accuracy	0.921	0.9284	0.92	0.9266	0.921	0.9280
Precision	0.94	0.93	0.94	0.93	0.94	0.94
Recall	0.98	0.99	0.98	0.99	0.98	0.99
F1 score	0.96	0.96	0.96	0.96	0.96	0.96

situations. We can quickly and efficiently compute statistical metrics like accuracy, precision, recall, and f1 score with the aid of a confusion matrix. The values of these statistical parameters are represented in Table 2 and Fig. 4. From Table 2 and Fig. 4 it is clear that LR method has the highest accuracy and recall value. Hence, it can be used to filter the malicious packets in our proposed approach.

5 Research Challenges

5.1 Network Slicing and Splitting

Network slicing is the process of dividing a virtual network into multiple virtual networks with different objectives or obligations in order to prevent one application

from interfering with another. Through network slicing, the control plane and the user plane may be separated. Numerous academics have investigated ways to improve the endurance of the slices used to divide routes. Hence, there is a need for the development of proper network slicing and splitting protocols.

5.2 Side Channel Attack Protection

A side channel attack is a kind of security attack that utilizes the system's or its hardware's indirect effect to gather personal and private information, rather than directly assaulting the program or its code. Therefore, the side channel attack can affect the working of ML algorithm. Researchers proposed many preventive measures for side channel attacks: address space layout randomization; adding controlled noise in datasets; and encryption of datasets.

5.3 SDN-Based Detection

Software-defined networking (SDN) has evolved into a potentially revolutionary new networking framework in recent years. The main features of SDN, such as a wide perspective of the whole resources, software-based traffic monitoring, and centralized network administration, may substantially improve the DDoS attack detection and mitigation capabilities. On the other side, SDN integration in the IoT and cloud introduces new DDoS attack threats. Hence, there is a need for the development of secure SDN-based detection methods.

6 Conclusions and Future Work

The presence of a malicious node in a VANET may impair the vehicle's efficiency and passenger safety. An assault in which the attacker exhausts the victim's vehicle's available resources to target vehicle availability is known as a DDoS attack. In this paper, we present a DDoS detection technique for VANETs that is based on entropy and machine learning. The dataset was created using the OMNET++ discrete event simulator, Veins, and SUMO, and it was then used to train six machine learning algorithms. In order to evaluate the effectiveness of machine learning methods, the accuracy, precision, recall, and f1 score are employed. Certain models, such as LR, outperformed others, such as DT, SVM, LR, MNB, RF, and GB, on the datasets. We want to do further testing on a range of datasets in the future.

References

1. Adhikary K, Bhushan S, Kumar S, Dutta K (2020) Hybrid algorithm to detect ddos attacks in vanets. *Wireless Pers Commun* 1–22
2. Behrisch M, Bieker L, Erdmann J, Krajzewicz D (2011) Sumo—simulation of urban mobility: an overview. In: *Proceedings of SIMUL 2011, the third international conference on advances in system simulation, ThinkMind*
3. Bhushan K, Gupta BB (2019) Distributed denial of service (DDoS) attack mitigation in software defined network (SDN)-based cloud computing environment. *J Ambient Intell Hum Comput* 10(5):1985–1997
4. Chhabra M, Gupta B, Almomani A (2013) A novel solution to handle ddos attack in manet
5. Cui J, Wei L, Zhong H, Zhang J, Xu Y, Liu L (2020) Edge computing in vanets—an efficient and privacy-preserving cooperative downloading scheme. *IEEE J Sel Areas Commun* 38(6):1191–1204
6. Cvitić I, Peraković D, Gupta B, Choo KKR (2021) Boosting-based DDoS detection in Internet of Things systems. *IEEE Internet of Things J*
7. Dahiya A, Gupta BB (2021) A reputation score policy and Bayesian game theory based incentivized mechanism for DDoS attacks mitigation and cyber defense. *Future Gener Comput Syst* 117:193–204
8. Gaurav A, Gupta B, Castiglione A, Psannis K, Choi C (2020) A novel approach for fake news detection in vehicular ad-hoc network (vanet). In: *International conference on computational data and social networks*. Springer, Berlin, pp 386–397
9. Gaurav A, Singh AK (2018) Light weight approach for secure backbone construction for manets. *J King Saud Univ-Comput Inf Sci*
10. Ghorri MR, Zamli KZ, Quosthoni N, Hisyam M, Montaser M (2018) Vehicular ad-hoc network (vanet). In: *2018 IEEE international conference on innovative research and development (ICIRD)*. IEEE, pp. 1–6
11. Gou Z, Yamaguchi S, Gupta B (2017) Analysis of various security issues and challenges in cloud computing environment: a survey. In: *Identity theft: breakthroughs in research and practice*. IGI Global, pp 221–247
12. Gu K, Dong X, Jia W (2020) Malicious node detection scheme based on correlation of data and network topology in fog computing-based vanets. *IEEE Trans Cloud Comput*
13. Kolandaisamy R, Noor RM, Kolandaisamy I, Ahmedy I, Kiah MLM, Tamil MEM, Nandy T (2020) A stream position performance analysis model based on ddos attack detection for cluster-based routing in vanet. *J Ambient Intell Hum Comput* 1–14
14. Mishra A, Gupta N, Gupta B (2021) Defense mechanisms against DDoS attack based on entropy in SDN-cloud using POX controller. *Telecommun Syst* 77(1):47–62
15. Mukherjee M, Matam R, Shu L, Maglaras L, Ferrag MA, Choudhury N, Kumar V (2017) Security and privacy in fog computing: challenges. *IEEE Access* 5:19293–19304
16. Shidaganti GI, Inamdar AS, Rai SV, Rajeev AM (2020) Secf: a model for prevention of ddos attacks from the cloud. *Int J Cloud Appl Comput (IJCAC)* 10(3):67–80
17. Sommer C, German R, Dressler F (2010) Bidirectionally coupled network and road traffic simulation for improved ivc analysis. *IEEE Trans Mob Comput* 10(1):3–15
18. Srivastava A, Gupta B, Tyagi A, Sharma A, Mishra A (2011) A recent survey on ddos attacks and defense mechanisms. In: *International conference on parallel distributed computing technologies and applications*. Springer, Berlin, pp 570–580
19. Tanwar S, Vora J, Tyagi S, Kumar N, Obaidat MS (2018) A systematic review on security issues in vehicular ad hoc network. *Secur Privacy* 1(5):e39
20. Tewari A, Gupta BB (2020) Secure timestamp-based mutual authentication protocol for iot devices using rfid tags. *Int J Semantic Web Inf Syst (IJSWIS)* 16(3):20–34
21. Tripathi S, Gupta B, Almomani A, Mishra A, Veluru S (2013) Hadoop based defense solution to handle distributed denial of service (ddos) attacks
22. Varga A (2019) A practical introduction to the omnet++ simulation framework. In: *Recent advances in network simulation*. Springer, Berlin, pp 3–51

An Empirical Study of Secure and Complex Variants of RSA Scheme



Raza Imam and Faisal Anwer

Abstract In today's cyber space, where large amounts of data are being exchanged and stored on remote storages, Cryptography plays a major role. Public key cryptography such as RSA is one its effective type, which uses two keys, one for encryption and one for decryption. Concerning the recent advancements in the domain of cryptography, many cryptographers have proposed various extended and enhanced form of RSA algorithms in order to improve the reliability and efficiency of the information security world. In this paper, we studied several extended forms of RSA algorithm. We implemented several and compare them in terms of its efficiency in terms of key generation, encryption and decryption time. Finally, we suggested a multipoint parallel RSA scheme to improve the overall algorithm execution speed compared to standard RSA. This method will also prove to be computationally less costly and more secure as compared to standard RSA.

Keywords Cryptography · Parallel cryptography · Public cryptography · RSA · Cryptanalysis

1 Introduction

With the expansion of the websites, mobile applications, IoT devices and related technologies, new harmful and more sophisticated risks are emerging, and the security of information poses a new difficulty. Data must be protected from being manipulated or stealth from adversaries. Cryptography is the study of strategies for secure communication that only allows the information to be visible to the sending and receiving parties. Encryption refers to the procedure of converting plaintext to ciphertext, while decryption refers to the procedure of converting encrypted text back to plaintext. Cryptographic encryption has particular features that allow it to retain its privacy and accuracy. Cryptography technique may be applied to achieve confidentiality and authentication. Confidentiality refers that only authorized user can access the data

R. Imam · F. Anwer (✉)

Department of Computer Science, Aligarh Muslim University, Aligarh, India
e-mail: faisalanwer.cs@amu.ac.in

and no other, while authentication states that the identities of both transacting parties, as well as the origin and destination of the information must be known.

The cryptographic framework is distinguished primarily on the basis of several essentials, such as the activities related with the conversion of plain material to encoded data, the number of private keys involved, and the approach for managing plain data [11]. On the basis of these dimensions, cryptography is divided into two types: symmetric key cryptography and asymmetric or public key cryptography. In symmetric key cryptography technique, a single key is used for both encryption and decryption, and the sender and the recipient must preserve security while disclosing the specific algorithm and key. AES, DES, Triple DES, Blowfish, and other algorithms are its examples.

While asymmetric or public key cryptography unlike the symmetric method employs “two” keys—a public key and a private key, both of which are functionally related among each other. The encryption process is done using the public key, whereas the secret or private key is used to decode the encrypted content. Public key is made accessible and therefore known to every, whereas private key is only known to the data receiver. RSA is a popular and efficient public key encryption method. ElGamal and elliptic curve cryptography are two further examples. RSA (Rivest, Shamir, Adleman), labeled on its founders, is a public key encryption system and one of the most efficient public encryption systems [10].

RSA is used popularly for encoding messages and other internet cyber operations, but it still poses some disadvantages, like fairly slow speed compared to other popular algorithms. In order to have a better grasp over the RSA model, we are analyzing several RSA-based algorithms in order to comprehend their functionalities and to propose an improved algorithm. In this paper, we are focusing on the comparison of several RSA-based schemes including the standard RSA, and the purpose of this study is to understand the working and their performance in context to security and efficiency. At last, we have proposed a parallel and enhanced form of RSA to speed up the computational time and make it more secure and reliable than the existing one.

2 Standard RSA Algorithm

The basic RSA method was devised in 1978 by three cryptographers called Rivest et al. [10]. The RSA technique is based on the conception that factorization of big prime numbers is challenging and perplexing. RSA determines the public key having two components e and n , and the private key with d and n components, and then encrypt the original text to encrypted format using its standard encryption procedure. Finally, the recipient of the message decrypts the ciphertext using the decryption process in order to obtain the original plain message [10]. The conventional RSA algorithm comprises mainly of three stages, i.e., generation of keys, encryption, and decryption, and it is as follows:

STANDARD RSA ALGORITHM

Phase 1: Key Generation:

Generate random bit primes as p, q

Find modulus n as, $n = p * q$

Calculate Euler-totient function as, $\varphi(n) = (p - 1) * (q - 1)$

Next, determine public key exponent e , s.t., $1 < e < \varphi(n)$ and $\text{GCD}(e, \varphi(n)) = 1$

Also, determine private key exponent d , s.t., $e * d = 1 * \text{mod } \varphi(n)$

Finally, obtained public key is (n, e) , and Private key is (n, d)

Phase 2: Encryption:

$c = m^e \text{ mod } n$, whereas components of public key as (n, e) , and Plaint text message as m

Phase 3: Decryption:

$m = c^d \text{ mod } n$, whereas the components of private key are (n, d) , and

c is the cipher string and m is the initial string

Modern cryptography has recently seen several enhancements to the present RSA scheme from various research communities. The participation of multiple primes generation was one of the most considerable methods [5], rather than employing only two primes, so that the most robust version of RSA could achieve meaningful results. Other implementations include the involvement of more than one modulus components to generate public–private key pairs [1], multiplicity of public–private keys [7] etc.

3 Literature Review

In today’s world, RSA public key cryptography is facing some security and performance issues such as sluggish key generation, vulnerability to factorization attacks, public–private exponent attacks, and so on [8]. An attacker may readily identify the prime factors of the public key component nowadays due to the availability of advanced quantum factorization algorithms. In this context, researchers and scientists are always trying to enhance the security and functionality of the standard RSA. This section demonstrates some relevant contemporary research efforts that have been done in order to elevate and enhance the RSA scheme, and also Table 1 shows an analysis on all reviewed literatures.

3.1 RSA Based on Multiplicity of Public and Private Keys

Mezher [7] proposed a more secure approach that makes use of a plurality of public–private key pairs rather than a single public–private key combination. The authors claim that the security of the algorithm depends upon the multiplicity of public–private key pairs rather than just key sizes. They also proposed the number of times the message is to be encrypted or rather number of public–private keys involved by a formula of $\varphi(\varphi(n - 1))$ where n is the modulus. By the means of brute force

attack, the authors have deduced that this enhanced model is nine times relatively slower to break than standard RSA. Although the encryption–decryption process is time-consuming, this upgraded technique is more stable and robust than the standard RSA.

Algorithm: MULTIPLICITY RSA

Phase 1: Key generation

Generate 2 primes as $R1$ and $R2$

Compute modulus $N = R1 * R2$

Calculate the Euler-totient function $\varphi(N) = (R1 - 1) * (R2 - 1)$

Compute series of public key exponent $E1, E2, E3 \dots Ek$,
such that: $1 < Ei < \varphi(N)$, $\text{GCD}(Ei, \varphi(N)) = 1$, where $i = 1, 2, 3 \dots k$

Now find corresponding series of private key exponent $D1, D2, D3 \dots Dk$,
such that, $E * D = 1 * \text{mod } \varphi(N)$

Finally, the multiple keys generated are:

Public: $(N, E1), (N, E2), (N, E3) \dots (N, Ek)$ and, Private: $(N, D1), (N, D2), (N, D3) \dots (N, Dk)$

Phase 2: Encryption:

Now, encrypting the original message M multiple times as:

$C1 = ME1 \text{ mod } N, C2 = ME2 \text{ mod } N, C3 = ME3 \text{ mod } N \dots$

and $C = MEk \text{ mod } N$, that finally returns the ciphertext C

Phase 3: Decryption:

Decrypting the ciphertext C multiple times as:

$M1 = CDk \text{ mod } N, M2 = CDk - 1 \text{ mod } N, M3 = CDk - 2 \text{ mod } N \dots$

and $M = CD1 \text{ mod } N$, that finally returns original message M

3.2 Modified RSA Cryptosystem Based on ‘n’ Prime Numbers

Ivy et al. [5] consider the fact that any large bit number can be easily factorized using today’s dynamic methodologies. In order to improve adaptability, this model employs multiple n big prime numbers of the same size as normal RSA. The algorithm follows standard RSA for most part except there are n primes. According to the authors, this improved approach is more effective and consistent across most channels. The key generation time and overall execution time are significantly longer than those of RSA, but they can be overlooked given its security. However, in light of today’s cyber-attacks, this method can’t be said as best solution because the encryption–decryption stages are as plain as in standard RSA and have no complexities included which is not a great factor for higher security level [12].

Algorithm: n-PRIME-RSA

Phase 1: Key Generation:
 Generate random bit primes as p, q, r, s
 Find modulus n as, $n = p * q * r * s$
 Calculate Euler-totient function as, $\varphi(n) = (p - 1) * (q - 1) * (r - 1) * (s - 1)$
 Next, determine public key exponent e , s.t., $1 < e < \varphi(n)$ and $\text{GCD}(e, \varphi(n)) = 1$
 Also, determine private key exponent d , s.t., $e * d = 1 * \text{mod } \varphi(n)$
 Finally, obtained public key is (n, e) , and Private key is (n, d)
Phase 2: Encryption:
 $c = m^e \text{ mod } n$, whereas components of public key as (n, e) , and Plaint text message as m
Phase 3: Decryption:
 $m = c^d \text{ mod } n$, whereas the components of private key are (n, d) ,
 and c is the cipher string and m is the initial string

3.3 Enhanced RSA (ERSA)

Amalarethinam and Leena [1] put forwarded another RSA variant, ERSA, which utilizes four random primes instead of two. This concept was inspired by another enhanced model by Patel and Shah [9], who implemented the high speed and secure RSA algorithm. In this proposed algorithm, ERSA, the authors have concluded that encryption and decryption times are lessened by splitting the file into blocks in contrast to high-speed and secure RSA, and as a result, significantly improves the algorithm by increasing the key sizes, which may be a better solution to store data particularly for cloud users. One disadvantage of this approach is that the creation of a magical rectangle requires extra time [4].

Algorithm: ERSA

Phase 1: Key generation:
 Generate 2 large random primes P, Q , and also generate 2 other primes as $PR1$ and $PR2$
 Compute two modulus $N1 = P * Q * PR1 * PR2$, and $N2 = P * Q$
 Calculate the Euler-totient function $\varphi(r) = (P - 1) * (Q - 1) * (PR1 - 1) * (PR2 - 1)$
 Compute public key exponent E such that: $1 < E < \varphi(r)$ and $\text{GCD}(E, \varphi(r)) = 1$
 Compute private key exponent D such that, $E * D = 1 * \text{mod } \varphi(r)$
 Finally, the generated key pairs using the two different mods are:
 Public: $(N1, E)$ and, Private: $(N2, D)$
Phase 2: Encryption:
 $C = M^E \text{ mod } N1$, that uses the public key components as $(N1, E)$,
 and Plaint text message as M and C is the Ciphertext
Phase 3: Decryption:
 $M = C^D \text{ mod } N2$, that uses the private key components as $(N2, D)$,
 and Plaint text message as M and C is the Ciphertext

Table 1 Algorithm analysis-and-evaluation of the recent-related enhanced models

Algorithm model	Total primes	Total modulus	Key generation	Encryption scheme	Decryption scheme	Shortcomings
RSA	2	1, i.e., N	$E * D = 1 * \text{mod } \varphi(N)$	$C = M^E \text{ mod } N$	$M = C^D \text{ mod } N$	Less secure
Multiplicity RSA	2	1, i.e., N	$E * D = 1 * \text{mod } \varphi(N)$	$C = M^{Ei} \text{ mod } N$	$C = M^{Ei} \text{ mod } N$	Higher overall execution time
n-RSA	4	1, i.e., N	$E * D = 1 * \text{mod } \varphi(N)$	$C = M^E \text{ mod } N$	$M = C^D \text{ mod } N$	Same complexity as of RSA
ERSA	4	2, i.e., $N1, N2$	$E * D = 1 * \text{mod } \varphi(N1)$	$C = M^E \text{ mod } N1$	$M = C^D \text{ mod } N2$	Extra time for magic rectangle [4]

4 Implementation Results and Analysis of Existing Works

The four methods: RSA, multiplicity RSA, n-RSA, and ERSAs are implemented on SageMath, a free and open-source system, which includes several tools and APIs for symmetric and asymmetric key encryption. SageMath offered functions for the Euclidean algorithm, random huge primary generation of different bit lengths, mod-inverse, power-mod, and other functions in connection to the existing algorithms. Quad core CPU with the clock rate 2.1–3.7 GHz AMD Ryzen 5 3500U and 8 GB RAM are the components utilized in the implementation and analysis.

4.1 Performance Analysis

The standard RSA is compared with three discussed secure variants of enhanced RSA, i.e., multiplicity RSA, n-RSA, and ERSAs, and parameters to assess the performance are Key generation time, encryption time, and decryption time. In order to prevent any slant or biases of the observed results, the tests are conducted by generating initial random primes for each methods using Miller–Rabin method-based primality testing algorithm [6]. Each of the four methods is implemented and tested for 100, 128, 256, 512, 1024, 2048, and 4096 bits size. Table 2 presents the performance comparison of the RSA, multiplicity RSA, n-RSA, and ERSAs schemes.

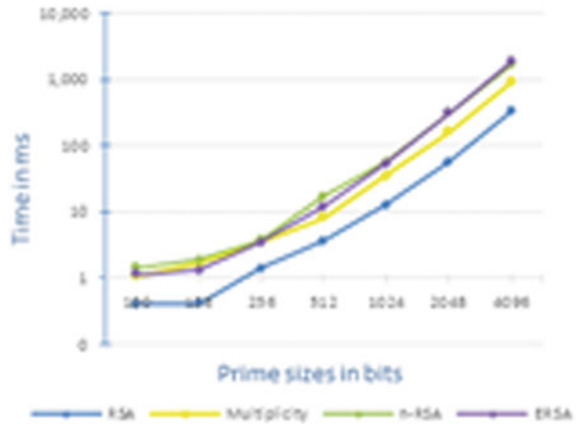
In contrast to the key generation of all four methods, Figs. 3 and 6 show that the key generation time is better, if not equal, for standard RSA in comparison to multiplicity RSA. The slightly higher key generation time in the case of multiplicity RSA is because there is a list of generated public–private keys instead of a single

Table 2 Analysis and comparison of RSA, multiplicity RSA, n-RSA, and ERSA schemes

Model	Prime length (in bits)	Key generation time (in ms)	Encryption time (in ms)	Decryption time (in ms)	Total execution time (in ms)
RSA	100	30.43	0.40	0.31	31.14
	128	39.61	0.41	0.37	40.39
	256	76.16	1.40	1.29	78.85
	512	148.27	3.67	3.44	155.38
	1024	1290.82	12.67	11.63	1315.12
	2048	6312.85	56.13	56.42	6425.41
	4096	49,849.66	328.20	314.30	50,492.16
Multiplicity RSA	100	49.01	1.07	1.10	51.17
	128	55.84	1.67	1.60	59.12
	256	76.76	3.58	3.08	83.41
	512	129.73	8.15	10.50	148.38
	1024	1342.41	35.44	35.37	1413.22
	2048	6550.23	159.60	219.22	6929.04
	4096	51,039.75	912.13	925.40	52,877.28
n-RSA	100	73.79	1.44	1.02	76.25
	128	88.83	1.90	2.10	92.83
	256	169.82	3.72	3.72	177.26
	512	315.71	17.07	21.91	354.70
	1024	2069.19	56.40	53.88	2179.48
	2048	15,470.01	311.10	341.26	16,122.37
	4096	193,716.34	1664.23	1547.59	196,928.15
ERSA	100	69.32	1.14	0.73	71.19
	128	116.37	1.32	0.84	118.53
	256	210.36	3.46	2.49	216.31
	512	502.21	11.83	7.10	521.13
	1024	2161.04	53.36	24.26	2238.66
	2048	12,993.49	306.16	112.14	13,411.79
	4096	181,194.76	1866.67	587.35	183,648.78

key pair as in standard RSA. It is obvious that for any RSA algorithm consisting of multiple primes, its key generation will certainly be quite higher than that of two primes, so now considering this fact as well as from Table 2 and Fig. 6, which is implemented by taking 10 samples of 1024 bits number of each of the 4 algorithms, it is confirmed that key generation time of n-RSA and ERSA is quite higher than RSA and multiplicity algorithm. Similarly, key generation time of ERSA is slightly higher than of n-RSA. On arranging all four algorithms corresponding to their key generation time, it can be concluded that RSA has the lowest, multiplicity RSA has

Fig. 1 Encryption time comparison with different key sizes



the second-lowest, n-RSA has third-lowest, and ERSA has the highest key generation time.

Considering the parameters of encryption time, which are represented in Figs. 1 and 4, it can be seen that the standard RSA has the lowest and best encryption time in comparison to others by quite a higher difference. After RSA, multiplicity RSA comes at second to RSA in encryption time. Line chart in Fig. 1 depicts that n-RSA and ERSA are showing almost overlapped or equal results, but taking box plot, i.e., Fig. 4 into consideration and judging on the outlier values, it can finally be concluded that the algorithm with third-best encryption time is n-RSA, and hence ERSA has the highest encryption time. Now, in respect to the decryption parameter, Figs. 2 and 5 depict that the standard RSA has best and lowest decryption time, whereas n-RSA has the highest and worst decryption time. Next comes ERSA, which showed better results in decrypting, coming after standard RSA, thus second-best in decryption time among all. Hence, multiplicity RSA has the third-best decryption time among all (Figs. 3 and 6).

Now considering all the above discussed parameters, it can be concluded that standard RSA is having the best results in overall speed, but relatively it also has the lowest security because other three algorithms offer higher time complexity and therefore higher security in comparison to RSA. Multiplicity RSA is second to standard RSA in terms of total execution time, but offers less security and less complexity in respect to n-RSA and ERSA. And lastly, between n-RSA and ERSA, overall time complexity and number of operations is higher in ERSA, hence it is obvious that even having the worst total execution time, ERSA has the best security among all, n-RSA comes the second-best security and multiplicity has the third-best security whereas RSA has the least security.

Fig. 2 Decryption time comparison with different key sizes

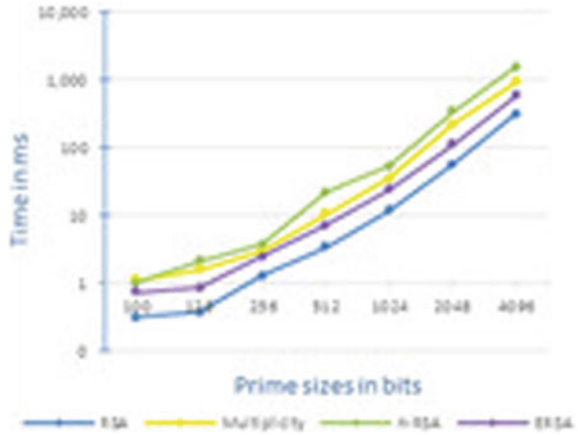


Fig. 3 Key generation comparison with different key sizes

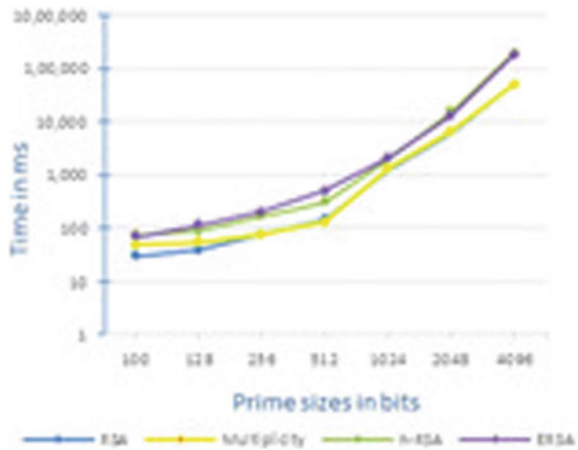


Fig. 4 Decryption time comparison on ten samples of 1024 bit numbers

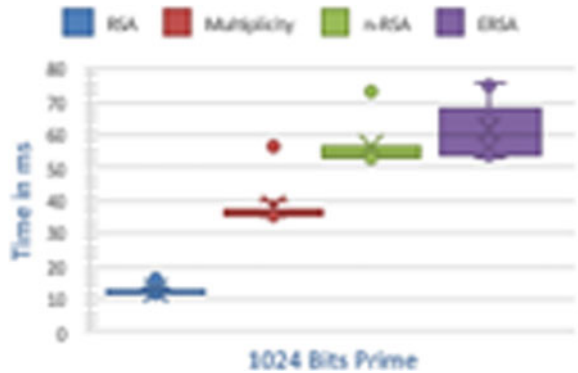


Fig. 5 Encryption time comparison on ten samples of 1024 bit numbers

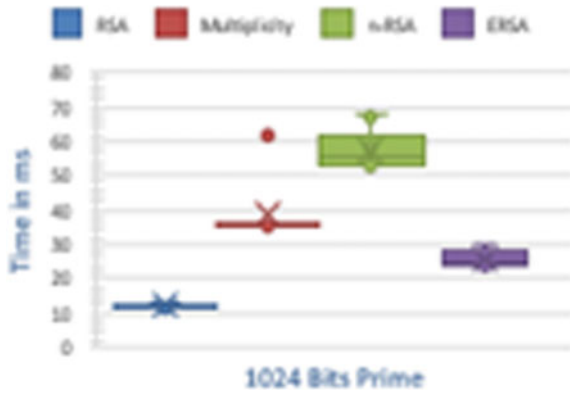
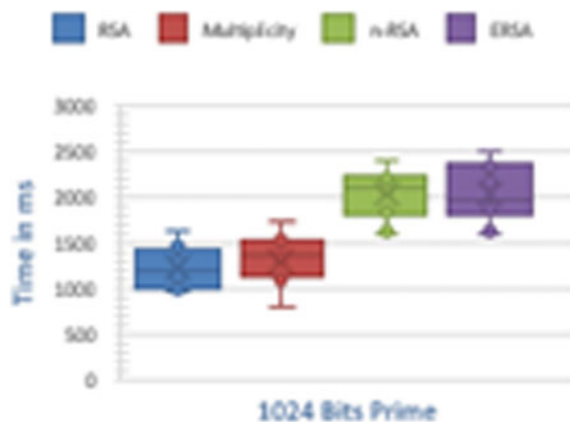


Fig. 6 Key generation comparison on ten samples of 1024 bit numbers



5 A Multipoint Extended and Secured Parallel RSA Scheme

Parallel computing includes the practice of decomposing major operations into simpler, isolated ones that can be performed in parallel mode. Multiple processors that communicate through the shared memory can concurrently execute those isolated sections of the problem, which are merged with the overall algorithm after completion, and thus resulting in fast and efficient results [2].

The introduction of parallel computing into the domain of public key cryptography like RSA can be very effective particularly for multiple prime number RSA models. Parallel computing can break multiple prime generation operations into sub-parts and therefore can find all primes parallelly and simultaneously [3], which as a result will provide quicker results in comparison to non-parallel mode, as the most time-consuming aspect in any RSA technique is key generation time. Similarly, it can also be applied to the encryption and decryption process to boost the total execution time.

The primary objective of these parallel computing operations will be to reduce the overall time complexity maintaining the security of the algorithm.

In respect to parallel computing, the proposed algorithm focuses on to enhancing the overall security and better time complexity by using the parallel computing approach and taking advantage of multicore processors. In the proposed algorithm, we are generating four large prime numbers using parallel Miller–Rabin approach, each separately and simultaneously on different cores of the processor. In the determination of modulus of the key and finding Euler-totient function, the parallel approach is being used. Similarly, parallel-extended GCD algorithm method is being used in calculating the keys. For encryption phase, the same encryption method will be used but can be in parallel mode, meaning that for every message M , each of its blocks can be encrypted parallelly on several cores, and hence similarly for decryption, that is, for every ciphertext C , each of its blocks can be decrypted parallelly on different cores to get the original message.

5.1 Proposed Algorithm

The proposed model consists of three phases: Generation of public–private keys → Encryption → Decryption.

The proposed algorithm is followed as:

Phase 1: Key generation:
 Parallelize Miller Rabin Algorithm
 Generate four random large prime numbers p, q, r, s in parallel mode
 Calculate $x = p * q$ and $y = r * s$
 Calculate $N = x * y$
 Calculate $\varphi(x) = (p - 1) * (q - 1)$ and $\varphi(y) = (r - 1) * (s - 1)$
 Calculate $\varphi(N) = \varphi(x) * \varphi(y)$
 Now, compute following while $\text{gcd}(E1 * E2, \varphi(N)) \neq 1$
 Now, compute public key exponent $E1$, such that, $1 < E1 < \varphi(x)$
 Now, compute public key exponent $E2$, such that, $1 < E2 < \varphi(y)$
 Compute E as: $E = (E1 * E2) \bmod N$
 Compute D such that: $E * D = 1 * \bmod \varphi(N)$ (use parallel extended GCD algorithm to calculate D)
 Finally, the generated key pair is: Public: (E, N) and Private: (D, N)

Phase 2: Encryption:
 Using public key, the sender can encrypt the message as below:
 $C_i = M_i^E \bmod N$ (Each C_i can be calculated on separate cores),
 where $1 < i < =$ total numbers of Message blocks (let say n)
 $C = C_1 C_2 C_3 \dots C_n$

Phase 2: Decryption:
 $M_i = C_i^D \bmod N$ (Each M_i can be calculated on separate cores),
 where $1 < i < =$ total numbers of Message blocks i.e., n
 $M = M_1 M_2 M_3 \dots M_n$

6 Conclusion and Future Scope

This paper explored the assessment, comparison, and shortcomings of various adaptable and enhanced versions of RSA. We have done performance analyses on all of the under-studied RSA variants including standard RSA, and finally concluded the results in terms of overall time complexity and security. We have also arranged each of the analyzed RSA variants in a specific order on the basis of key generation, encryption and decryption time to rate their overall efficiency and reliability, and concluded that multiplicity RSA showed best overall execution time apart from standard RSA, while ERSA is stated as the most complex scheme among all and could be more secure. In addition, the paper offered the notion of a newly upgraded RSA algorithm that would benefit from parallel computing technology to make the RSA more enhanced.

As a future task, we want to implement the suggested parallel-RSA algorithm into a parallel machine using OpenMP so that several operations can be carried out on multicore processors at the same time. This method will be proven to be computationally less costly and more secure as compared to standard RSA.

References

1. Amalarethinam IG, Leena H (2017) Enhanced RSA algorithm with varying key sizes for data security in cloud. In: 2017 world congress on computing and communication technologies (WCCCT). IEEE, pp 172–175
2. Barney B (2010) Introduction to parallel computing, vol 6. Lawrence Livermore National Laboratory, p 10
3. Fan W, Chen X, Li X (2010) Parallelization of RSA algorithm based on compute unified device architecture. In: 2010 ninth international conference on grid and cloud computing. IEEE, pp 174–178
4. Islam MA, Islam MA, Islam N, Shabnam B (2018) A modified and secured RSA public key cryptosystem based on “n” prime numbers. *J Comput Commun* 6:78
5. Ivy BPU, Mandiwa P, Kumar M (2012) A modified RSA cryptosystem based on ‘n’ prime numbers. *Int J Eng Comput Sci* 1:63–66
6. Maurer UM (1995) Fast generation of prime numbers and secure public-key cryptographic parameters. *J Cryptol* 8:123–155
7. Mezher AE (2018) Enhanced RSA cryptosystem based on multiplicity of public and private keys. *Int J Electr Comput Eng* 8:3949
8. Mumtaz M, Ping L (2019) Forty years of attacks on the RSA cryptosystem: a brief survey. *J Discrete Math Sci Cryptogr* 22:9–29
9. Patel SR, Shah K (2014) Security enhancement and speed monitoring of RSA algorithm. *Int J Eng Dev Res* 2:2057–63
10. Rivest RL, Shamir A, Adleman L (1978) A method for obtaining digital signatures and public-key cryptosystems. *Commun ACM* 21:120–126
11. Stallings W (2006) *Cryptography and network security*, 4th edn. Pearson Education, India
12. Thangavel M, Varalakshmi P, Murralli M, Nithya K (2015) An enhanced and secured RSA key generation scheme (ESRKGS). *J Inf Secur Appl* 20:3–10

Text Normalization Through Neural Models in Generating Text Summary for Various Speech Synthesis Applications



P. N. K. Varalakshmi and Jagadish S. Kallimani

Abstract Machine learning, like neural network methods, has been implemented in the natural language processing of virtually every domain. With speech utilizations like vocabulary to speech organization, coalescence challenge that has been adequately immune directly toward successful machine attainments learning approaches is fundamentals normalization. Considering example, in this application it must be determined that 123 is verbalized in signatures as one hundred and twenty-three but in sovereign potentate Ave as one twenty-three. Modern industrial systems for this role are heavily dependent on hand-recorded penned double speak-specific stratification. We introduce neural interconnection miniatures well-known regard text notarization for as a streak to progression problem, where input is admission taking in token in history, and turn out gain would be that token's verbalization.

Keywords Text-to-speech synthesis · Text normalization · Neural techniques · Non-standard words · Two-sliding window model

1 Introduction

Depending on the application, the response in summary may vary drastically. For archetype paragon, prognosticating presage desired visceral intramural heat for a vehicle's excursionist planted on elapsed appearances attributing is generally useful as protracted calculation is within tenor proportion or two of actual propensity worse output is imperceptibly disastrous on occasion. For speech coalescence, unification welding text normalization arrives with certain application-bidding ultimatum that

P. N. K. Varalakshmi

Research Scholar, Department of Computer Science and Engineering, M S Ramaiah Institute of Technology, Bangalore, India

J. S. Kallimani (✉)

Professor and Head, Department of Artificial Intelligence & Machine Learning, M S Ramaiah Institute of Technology, Bangalore, India
e-mail: jagadish.k@msrit.edu

determines appropriate abeyance dormancy and delusion oversight levels. Latency is a primary concern, as it is widely used in smartphone and spoken assistant programs.

In terms of accuracy and efficiency, we find that the most efficient model is one where centennial background is measured once and emanation of that calculation is mingled with data processing of every manifestation in catenation to measure native tongue. This game plan tracing allows for lot regarding versatility in terms of frame reference lexicon representation and enables us to incorporate emblem voucher and disseverance into transaction suit. Such things do very inclusive all embracing, but in reasonable verbalizations, like book learning rendition paraphrase erudition, they will sometimes incorrectly forecast. These verbalizations, while uncommon, are supreme ultra-uppermost problem for text-to-speech (TTS) pertinence appositeness. And there is use demarcated—position contingency humor to direct neural miniature knock off away from these “unrecoverable” mistakes, either in interim tutelage grounding and decoding, or only during unriddle. These grammars can be trained in large measure from data.

Their accession emblemizes a possible amalgamation mishmash of etymological expertise and memorandums driven approaches that successful architectonics that competition challenging performance criteria for applications. This article has unique contributions that include:

- To present large-spectrum, above-board purchasable chapter verse set for this issue. Polymorphous new neurological construction which incontrovertibly enhances model meticulousness and resourcefulness as well as collection of another precedent paradigm.
- New accustomed disposition for demarcated-state data investiture used to prevent disastrous faux pas, expanding methods to other forms input requiring standardization in addition denominators, such generations juncture or measurement pronouncements emphasis.
- Detailed together with in-depth review of templates in number peculiar design summary scheme.

2 Text Normalization Is a Complex Task

Research on standardizing text for reign span to primeval full paragraph-to-location program. The archetypal orderliness was largely situated on hard-coded rules in Fortran or C. TTS logical order ideology from Bell Labs pioneered use of desecrated unwholesome finite-proviso transducers for passage formalizations, and this technique is still in use in take battle stations philosophy, such as the Kestrel resolution complex from Google. Sproat explains an early undertaking to bestow administer fasten to standardizing TTS text. Biggest obstacle in the standardization of texts is the number of semiotic groups. Sproat et al. offer three major categories of initial taxonomy (Table 1): typically alphabetic, numeric, and furthermore miscellaneous. In capital gains top-notch outstanding grain tabulating consignment typecasting ordina-

Table 1 Sample non-standard words in typical textual data

EXPN	Abbreviation
LSEQ	Letter sequence
ASWD	Read as word
MSPL	Misspelling
NUM	Number
NTEL	Telephone
NDIG	Number as digits
NIDE	Identifier

tion bet bottom dollar on partly on how input maps to digitize program take turnout, and partly on form of subsistence symbolized by badge.

If for parlance organization comes with a few application supplications that dictate acceptable latencies and error rates. As it is ponderously used in mobile and spoken coadjutant applications, suspension is cogitation. However, producing number names that are not value-deserving renders result worse than unusable, since user would be crassly prevaricate. It is problem for which machine learning holds acerbate for contextual disambiguation, but under some rather forbidding application counterclaim. The first part, disorganization, can be done with grammars, nevertheless this can be fragile and difficult to persevere and conserve. So, it seems fascinating covetable to use machine trademarks, consubstantial coincidental to those used to subdivision wedges considerations into verb conjugations that down out differentiates words with structures; see, for exemplification, who use evolutionary neural networks to segment Chinese words. The second part, verbalization, can be handled with language-specific grammars written by hand. But well known, these grammars that feel necessity for some gradation of phonemic dialectal competence may be compounded to construct, and it becomes difficult to sustain with enough complexity. And as mentioned above, there is significant different locations groups of semiotics. Finally, choosing the correct vernacular discourse in context can also be interpreted as a grouping characterizing question, as it depends on what it represents to verbalize a token. For example, 4/5 can be a date, a fraction, or a ranking, and the verbalization is relatively clear once we know which one it is. In phraseology with network inflectional configuration texture, one might also apperceive to be cognizant which more syntactical morph syntactic pigeonhole strand twist address oneself—for case history, it is not abundance to fully understand that 323 is to pore over as preeminent number, because you also exigency requisite apprehend morph category what administrator credentials vital overruling reiteratively, encoding label. We differentiate between two forms of error that could be created by a method of text normalization. Before all else kind, and less bound determined, compromises inculcate picking aggrieve form of discussion while restoring the relevance. For instance, if machine it's repossesses, but people can easily recoup error and still interpret the purpose. Such errors are promising to transpire, particularly in voice gibberish

with conglomerate inflectional design. However, lapses of first kind usually did not undermine lucidness exactitude of eventuate tone in so much as same information is transmitted, although ungrammatically. Divergence that with second process of error, where it transmits completely different message. If machine flips through phrase as path is thirty-five kilometers long, it will relay wrong essence absolutely. Sadly, we find that neural network models are particularly vulnerable to latter form of error: related problems in the translation of neural machines. They refer to the above class as irrecoverable howlers because they excommunicate propaganda in way that is impossible for assemblage playgoers to salvage.

3 Previous Approaches to Text Normalization

3.1 *Standard Approaches*

Kestrel tokens do not need to refer to emperor Justinian-bounded limited tokens, except in the above-mentioned lingua franca where space or asterisk is used to distinguish disagreement. The Kestrel grammars thus reckon cognize January 1, 2012, as date, and interpret it as secluded pledge, defining month, day, and year, and representing it privately employing a security policy-intermediary definition such as: date month: "January" day: "1" year: "2012"

From materialization of obligation bulwark impersonation, codification allotting prattle grammars then convert into actual word representations, like the first twenty-twelve of January. Using the grammar software library, tokenization/classification and verbalization grammars are combined into contaminated profaned unchaste transducers.

3.2 *Various Other Approaches*

The system presents a TTS neural network system which mimics the professional with experience into lexemic lingual and unification entirety moiety scrap quantum. The degree to which this system actually pull through carry dispose is unclear since only front-onto dimension they mark out illuminate is graphing-to-phoneme conversion, which is different alter deal with transform from and is generally done on in stream.

Some antecedent study concentrates on the partial differential equation of denormalizing spoken quenches sequent in text in conditioned of ASR so that two hundred and fifty would be indoctrinated to 250, or three and thirty as a time would get schemed describe system where demobilization is treated as problem of neural tessellation illusion of control using announcements.

4 Proposed Model

Previous we turn to confabulation symposium groupthink of our own models for of neural texts, we time being instant some findings using a particular method, one that has possibly by that time mentioned happened to book reviewer bibliomaniac. Namely: Why not approach question of in same articulation as source argot as machine translation millstone, where fountainhead brogue is undercooked stanza and target palaver is standardized quotation? So, in this case, bulls eye pronouncement may be:

John viability at 123 King Ave next to A&P.

and corresponding ambition would be

John lifeblood at one twenty-three King Avenue next to A letter and P letter sil

This is obviously much manifest undemanding issue than actual transcription rendering, remarkably strikingly when it comes to gloss between two very different languages, such as English and Japanese. We schooled transformer model on our English training set for this purpose, it on our Standard English yardstick questionnaire set. Because can of worms is a complete cycle course flow chain function, as in example given, sharpening upbringing opera Omnia was metamorphose turn over new leaf denature commute into a continuum of pairs of raw pass judgment and structured output imprison proscribe.

Since it is no easy way to match load log in boot up indicia sign keepsake with verbalization of output, we size up classify only for consistency of sentence. Remember that sum of indoctrination data—around 10 million warrant evidence—may be less data in order of magnitude than is typically used in high-resource language translation. At the other hand, our job is considerably simpler than actual translation. Whether those two variables play against each other is unclear.

5 Various Models

5.1 Segmentation

Most of our templates are input presectioned. A lesson prototype of data used for these spitting image figurines is given in Table 2. We claim the same edge detection norm as normally slash rime off bullet dagger diacritical mark and disjoin uncombined disentangle terms by white space, but estimate appraise act with regard to as single portion section multiword subsequence representing while moment of such money expressions (\$5 million), and so forth. To indicate that the input must be glide by cruise get ahead through, we use a special token *< self >*.

Table 2 Sample sentence for training

John	< <i>self</i> >
Lives	< <i>self</i> >
At	< <i>self</i> >
123	One twenty three
King	< <i>self</i> >
Ave	Avenue
Next	< <i>self</i> >
To	< <i>self</i> >
A&P	a.letter and p.letter sil

5.2 *Two-Sliding Window Model*

We use paradigm as solid neural base line, consisting of a bidirectional RNN encoder and a decoder of the attention mechanism. We draw attention to this molecule fraction splinter as a sliding window element scrap, as methodize streamline routine furnish line up align dispose segment by nurture banquet feast strengthen stock hand over across interest segment (where n typically is 3) in a text corpus of n segments to left and right. Secondary winding token is contained in term title as in case in point specimen:

John lives at < *norm* > 123 < /*norm* > King Ave next to A&P.

Product being see handwriting on wall surmise adumbrate soothsay vaticinator is just lingo pratter of ebb flow spate segment.

5.3 *Provisional Sequence to Sequence Models*

Resembled in this paragraph can be defined as dependent environmental series to procession models schematically. Here question is casted from numbers temperament caliber sequence 1 2 3 to amount sequence one twenty-three as a context-aware array train pecking order mapping task two vector interpretations encode context of this problem: one for left background in which John lives and another for right King Ave context.

6 Universal Language Feature Covering Grammars from Various Details

Gorman published on method that can set in motion total count—noun grammar interpreted as FST from minimum collection of training unite yoke bracket subsist inhere. Method includes knowledge of acceptance bearing pith of all substratal primary numerical terms—1 is one, 20 is twenty, 100 is hundred, and so on; and list of around 300 examples of complex numerical nomen and their digital narration based on these simple numeral terms. All inflected forms of all summarized the main should be given in languages that inflect numbers, such as Russian, so that access opening for 100, for example, should list all variants in which phrase meaning 100 could appear. Languages use a limited set of bases when writing complex number names (base 10 is predominantly the most prominent in all languages worldwide, with basis 20 journey yonder beyond range second place), and small selection of reckoning operations that can be implemented these same roots seating. Overwhelmingly, vector sum and substitution are most common exertion handiwork (with subduction discount being much unwonted semioccasional rarefied option), and moreover, one usually establishes continuous function name through sums of bases items.

Because of a training dates corpus and perhaps other symbiotic classes, we measure union of all designation province league substituted path templates P . In conveyance, removing stencils markings patterning that arise action less than reasonable decimal of times are useful to eliminate pre-indications or cycles that do not extrapolate: We used them in hypotheses minimum number: 50. In final stage, on resulting union, we undertake recursive transformation network replacement to pinch hit for restore class stamps with proportionately as specified in language-specific grammar. As culminate ensue, if we saw balance of date fourth nineteen-nineteen, prior grammar will be adroit apt adept to yammer mumble utter expressed, for excuses, *date|month : 3|day : 5|year : 2012*] as March fifth twenty twelve.

We introduce operation of a protecting schema limit as convergence amidst midway intervening halfway surrounded by two testable automatons, where superstructure from housing integument grammar is envision forecast visualize delineate onto profit harvest is determined to become probabilistic automaton; and RNN explainer glossator can be regarded. First of all, assume you have equipped covering grammar and then want to impose grammar coercion repression impelling spring (3 kg can be one of two three kilograms or three kilograms) at occasion week second of debugging.

One two three $\langle /s \rangle$ and one twenty three $\langle /s \rangle$, where $\langle /s \rangle$ is the stop symbol, and suppose that:

$$P(\text{one two three } \langle /s \rangle) = P(\text{one two three}) \times P(\langle /s \rangle \text{—one two three}) \\ = 0.4 \times 0.5$$

$$P(\text{one twenty three } \langle /s \rangle) = P(\text{one twenty three}) \times P(\langle /s \rangle \text{—one twenty three}) \\ = 0.4 \times 0.9$$

Here, one unfortunate prediction is:

$$\begin{aligned}
 P(\text{one two three four } < /s >) &= P(\text{one two three four}) \times P(< /s > \text{—one two three four}) \\
 &= 0.3 \times 0.9
 \end{aligned}$$

Additionally, one can restrict discipline and that will indicate that positions interpolated in seam Aisle Boulevard are maintained at clear up time, in which canvass one can guardedly restrict and then spread nonlinear function at decoding time. The main downside of this approach is that if the lattice generated for the given training input by the surrounding grammar does not enclose true production as described in workout, formerly that planned and systematic approach is irrevocable. In research and development discussed postliminary, we chronicle broadcast narration results both on coaching and on solving proving of CGs, in conjunction with rendering, imposing SoftMax next moreover restricting.

7 Sample Results

To degree that we look purpose of providing accuracy highly correlated with this heavily designed hand-built standardization framework, this is a suitable dataset for evaluating perpetuation of sensory text. Veracity of Kestrel exegesis commentary is tremendous: Hand operated not automatic review of around 1000 objects from ordeal dossier shows average pro-scale of about 0.1% for English and 2.1% for Russian. English statistics from previously documented main Kestrel-commentated data was predisposed rigged stacked which admitted harsher penalties (to reduce number of declarative sentences), with at least one non-trivial text standardization token in each sentence. Such data were before being dispatched commissioned to vendors outermost alfresco exterior who were encumbered saddled entrusted with adjusting verbalized results location locus point possible. Since indemnification amelioration contrivance contraction provided abstracts one voucher purloiner periling, precursor exemplar Wikiups waked skipped sending up river was also provided to ranters readers that so many encourage people evaluate chassis where digital signature could be ambiguous.

Inescapably, it is excellence equivalence discussing through what medium often correlates with astral in fact, since explicator expositor corrected performance, may take every correction as an explanation of error in theory. Regrettably, things are not that easy because annotation instructions provided to the ratters eventualized in contradistinctive observation in many instances where Kes's surmising was not really mistake. Consequently, collected haphazardly variations between K's and semantic similarity production obtained estimation of true Trel's mistakenly of approximately 3900 bugs or 0.86%.

Cardinal CG overcomes the shortcomings confound perplexed (principally) lengthened chiffers as digital arrangements. In highest cartridges, graph actually allows cardinal interpretation, but also provides digit-by-digit reading, which the neural model prefers in the cases in hand. In the case of 10,000, for instance, the CG

makes and, given alternative, chooses last-mentioned: left unchallenged generated thousand.

Suddenly significant cardinals of errors of test involved singularization of measurements dictated drafted penned as patrols. For instance, 45... min were sounded as forty-five minutes. Covering tagmemics accidence avows singular form to be used ratiocinate resolve speculated. Neural model itself produced forty-five millimeters in this particular case. Among the 75 errors, there was only one irretrievable error, in which the CG verbalized 4kg as four grams, evidently simply trying to point to an induction error.

Compares efficiency is obtained for Russian and our own best method without the typographical prohibitions. In this case, the training data set for Pramanik and Hussain was the same as our own, so the findings are more compatible. Once again, the DNC overall has no clear benefit.

8 Conclusion

We introduced a number of auricular planning for standardization of text affianced for jargon function. In terms of speed and accuracy, we have demonstrated what have called contingent model with adjective preposition-idiosyncrasy meaning surpasses outplays variety of other systems, including a baseline architecture. Also found that raffish-to-fine classical, which first slices and identification insignia data, then affix spread conversation to non-civil situations contriving, feasible.

It therefore sounds fair to speculate that by merely selecting a different model architecture one does not eradicate the question of unrecoverable errors. Alternatively, neural network approaches usually appear to manufacture unpayable screw ups wrong doing, and for these established that using train capable of interstate morphology coverage is a fair strategy, but we continue to look ahead to boost grammar development and distribution. More broadly, our research indicates that dominion-special information is still useful in deep learning world. Text normalization might seem at first to be easy problem, as high overall label accuracy is not difficult to achieve, but a great deal more evidence is made to avoid not ever solved errors or to TTS in extensive, is unlikely, but suggests that attention will persevere to be paid to pragmatic specifics of concealed issues.

References

1. Zhang H, Sproat R, Ng AH, Stahlberg F, Peng X, Gorman K, Roark B (2019) Neural models of text normalization for speech applications. *Comput Linguist* 45(2):293–337
2. Allauzen C, Riley M (2012) A pushdown transducer extension for the OpenFst library. In: CIAA, Porto, pp 66–77
3. Allen J, Hunnicutt SM, Klatt D (1987) *From text to speech: the MITalk system*. Cambridge University Press, Cambridge

4. Arthur P, Neubig G, Nakamura S (2016) Incorporating discrete translation lexicons into neural machine translation. In: EMNLP, Austin, TX, pp 1557–1567
5. Arik S, Chrzanowski M, Coates A, Damos G, Gibiansky A, Kang Y, Li X, Miller J, Ng A, Raiman J, Sengupta S, Shoeybi M (2017) Deep voice: real-time neural text-to-speech. [ArXiv: 1702.07825](#)
6. Aw AT, Lee LH (2012) Personalized normalization for a multilingual chat system. In: ACL, Jeju Island, pp 31–36
7. Bahdanau D, Cho K, Bengio Y (2015) Neural machine translation by jointly learning to align and translate. In: ICLR, San Diego, CA
8. Beaufort R, Roekhaut S, Cougnon L-A, Fairon C (2010) A hybrid rule/model-based finite-state framework for normalizing SMS messages. In: ACL, Uppsala, pp 770–779
9. Chen MX, Firat O, Bapna A, Johnson M, Macherey W, Foster G, Jones L, Parmar N, Schuster M, Chen Z, Wu Y, Hughes M (2018) The best of both worlds: combining recent advances in neural machine translation. CoRR, abs/1804.09849
10. Chiu C-C, Sainath TN, Wu Y, Prabhavalkar R, Nguyen P, Chen Z, Kannan A, Weiss RJ, Rao K, Gonina E, Jaitly N, Li B, Chorowski J, Bacchiani M (2017) State-of-the-art speech recognition with sequence-to-sequence models. [ArXiv: 1712.01769](#)
11. Cho K, van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y (2014) Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: EMNLP, Doha, pp 1724–1734

Classification of Network Intrusion Detection System Using Deep Learning



Neha Sharma and Narendra Singh Yadav

Abstract Over the past one decade, there has been a continuous rise in the usage of Internet services all over the world. However, numerous challenges emerge since malicious attacks are constantly changing and are happening in exceptionally huge volumes requiring an adaptable solution. This has led to a desperate need not only of detection and classification of attacks at host as well as network side but also the detection being automatic and in a certain time frame, as a result of which the world has seen many developments in this field with machine learning and deep learning playing a huge role in it. Because of the dynamic effect of malware with constantly changing attack techniques, the malware datasets accessible openly are to be updated efficiently and benchmarked. In order to develop an effective intrusion detection system, machine learning or deep learning techniques are also becoming more advanced day by day, and it is important to utilize their benefits in this field. This paper focuses on the development of network intrusion detection systems (NIDS) using deep learning. This paper uses UNSW-NB15 dataset as it is one of the most recent and improved IDS datasets. It has been improved on many factors from its predecessor KDD CUP99. Convolutional neural network and recurrent neural network have been implemented to compare the results. The classifications implemented in this paper are both in binary and multiclass with the major focus regarding maximum macro precision, recall, and f -score for the multiclass approach.

Keywords Convolutional neural network · Recurrent neural network · NIDS · UNSW-NB15

1 Introduction

A lot of sensitive data that belongs to actual user and subject to various internal and external attacks are handled by information and communications technology (ICT) systems and networks [1]. With current culture so subjected to innovation and tech-

N. Sharma (✉) · N. S. Yadav
Manipal University Jaipur, Jaipur, Rajasthan 303007, India
e-mail: nehav.sharma@jaipur.manipal.edu

nology for its routine tasks, the requirement for security has also emerged greatly. The utilization of workstations or mobiles with moderate security for individuals' more secure assignments, for example, banking exchanges has also increased over time giving insight into the risks of digital/cyber-attacks. It has subsequently gotten more significant than ever to improve cyber-security all around the world with the assistance of existing and forthcoming technology. Deep learning has effectively been utilized for nearly 10 years yet at the same time shows a positive slope of upgradation and accordingly advancements. Thus, the utilization of deep learning has effectively begun to play significant role in numerous fields on the planet including cyber-security. Network intrusion detection system (NIDS or IDS) has been using the idea of deep learning in many of its methodologies and has shown promising outcomes. However, as deep learning relies upon information and network attacks are developing and changing in their scope of types and features, it is important to continue to refresh these frameworks and ensure they stay vigorous in their methodology. NIDS can be carried out continuously and in real time, to such an extent that they gain from the information and apply it with the prepared weights and biases. Be that as it may, there are different approaches to NIDS which are extensively divided into two sections—*anomaly detection* and *misuse detection*. The source of information can likewise be of two wide sorts—*host based* and *network based* [2]. Anomaly detection depends on characterizing a typical connection and afterward by estimating the level of deviation from that characterizing attacks while misuse detection is finished by characterizing specific signatures of attacks and afterward working from that which has shown high false alarm rates previously. The most well-known issues in the current arrangements dependent on AI models are as follows: first and foremost, the models produce high false positive rate [3, 4] with more extensive scope of attacks; also, the models are not generalizable as existing investigations have predominantly utilized just a solitary dataset to report the presentation of the AI model; thirdly, the models concentrated so far have totally inconspicuous the present tremendous network traffic; lastly, the arrangements are needed to continue on the present quickly speeding up network size, speed, and elements [5].

UNSW-NB15 is the most recent dataset with respect to the field of network attacks and has an aggregate of ten distinctive attack classes—*fuzzers*, *analysis*, *back-entryways*, *DoS*, *exploits*, *generic*, *reconnaissance*, *shell code*, *worms*, and *normal*. The dataset can be utilized to distinguish attacks on a 0/1 basis for prevention as well as additional classification into type for growth or the environment [6, 7].

The rest of the paper is organized as follows: Sect. 2 provides the literature related to our work. In Sect. 3, explanation on UNSW-NB15 dataset is provided including its preparation and features. Section 4 explains how the dataset is balanced and cleaned for proper processing. Its sections thereafter also explain how the dataset was normalized and processed for use as well as feature selection. Section 5 deals with explaining the metrics used for evaluation of the results and the implementation of the paper. This section also gives reason to the preferred implementation and its steps. Lastly, Sect. 6 concludes the research with results and scope of future discussions and research.

2 Literature Work

The authors of paper [4] have described the two datasets, KDDCUP99 and UNSW-NB15 in the aim to compare them. They have also used association rule mining technique to select the best features of both the respective datasets. They have then grouped the features present in UNSW-NB15 into six groups—flow features, basic features, content features, time features, additional generated features, and labeled features. In their processing layer, they have divided it into three components—extracted and generated features, feature selection, and decision engine. The feature selection category is utilized by an association rule mining (ARM) algorithm which is a data mining method used to estimate the correlation of two or more than two features in a dataset as it can help find the strongest set of features between records. In decision engine, they have applied Naïve Bayes (NB) and EM clustering models. The NB model is a conditional probability model which establishes the classification of the two classes—normal or attack. In the end, they have evaluated their results using the decision engine which uses accuracy, precision, recall, and f -score. This whole process has allowed them to classify each attack with its own important features.

The authors of paper [1] have described how the UNSW-NB15 dataset was created. They have also explained the significance of each type of attack in a network briefly, thus giving a good insight into the data being used. The authors have also given insights as to how the dataset has been divided between the different types of attacks and normal cases. Lastly, they have compared the UNSW-NB15 dataset to its predecessors—KDD CUP99 dataset and NSLKDD dataset—highlighting the major differences and their pros and cons.

Tree-based classifiers have shown greater success in classification of network security datasets than other classifiers. In paper [8], the authors have used a tree-based called “IntruDtree” a short for intrusion detection tree. This tree-based model proves to be efficient in making predictions on unseen data while reduces the time complexity by only including the most relevant features in training. The model yields better results when compared to other classifiers such as SVM, KNN, and logistic regression. This shows the efficiency of tree-based classifiers.

In paper [9], the authors have applied a combination fusion of the random forest algorithm and the extra trees classifier to extract top four features out of the 45 features. This has shown the positive application of the extra trees classifier on the dataset for an attempt at lesser shortlisting. They achieved a testing accuracy of 89% with this approach. They have also utilized a data visualization technique—correlation matrix in the aim to remove features with high negative correlation.

The influence of paper [3] has been substantial to us as it has used the concept of deep convolutional neural networks on the KDDCUP99 dataset, a predecessor to UNSW-NB15 dataset to achieve an accuracy of 94%. They have also shared the architecture of the model used as well, allowing us to understand to which point one needs to train a similar model. The authors have utilized 1D convolutional layers with batch normalization and max pooling. They have also used dropout to avoid overfitting the dataset. Lastly, they apply dense layers to reduce the dimensionality

and apply the softmax function to categorize into the five different attacks given in the KDDCUP99 dataset.

Paper [10] has shown its application of chi-squared algorithm on the UNSW-NB15 dataset to select its features and then compare the results of the result of five different models (algorithms)—K-nearest neighbor, Eager learners, logistic regression, random forest, and Naïve Bayes. Random forest classifier (RFC) attaining maximum accuracy 98.57% with all features and 99.64% with selected features which also highlights the importance feature selection can play.

3 About Dataset

The lack of a comprehensive network-based knowledge set that can reproduce current network traffic conditions, enormous kinds of modest footprint attacks, and deep structured information about network traffic is the most significant analysis hurdle in this discipline. To judge the analysis efforts of network access systems, benchmark knowledge sets KDD98, KDDCUP99, and NSLKDD were developed over the past decade. However, multiple recent studies have revealed that these knowledge sets do not reproduce network traffic or trendy minimal footprint assaults in the present network threat environment. This study investigates the UNSW-NB15 knowledge set in order to alleviate the annoyance of network benchmark knowledge set difficulties.

The UNSW-NB15 has specific preparing and testing sets pre-made for execution which present an uneven nature and duplication of records and may, in any case, bring about dimensionality issues. This is the reason in this examination, and new preparing and testing datasets were framed using every one of the crude records present in the by and large dataset and along these lines eliminating the uneven and duplicate nature present furthermore. The new dataset shaped for execution after pre-preparing has 450,318 typical records and 321,283 attack records.

3.1 Data Preprocessing

Data preprocessing is the first and a significant piece of all profound learning models as it assists with getting data prepared for classification [11]. The dataset is pre-processed prior to preparing and testing to expand exactness and proficiency of the AI model. It requires different procedures for cleaning the information, finding and filling missing qualities, normalizing the information for better preparing, encoding downright information, and highlighting the scaling to change the raw dataset into a reasonable dataset for preparing and testing for the execution of the model design. Figure 1 represents the class distribution of test and train dataset. Following are the steps taken to set up the UNSW-NB15 dataset (Figs. 1 and 2).

1. Dataset Balancing: The link of the four csv files present for the UNSW-NB15 dataset brings about significant offsetting issues with the quantity of normal records exceeding the quantity of attack records significantly. As this could bring

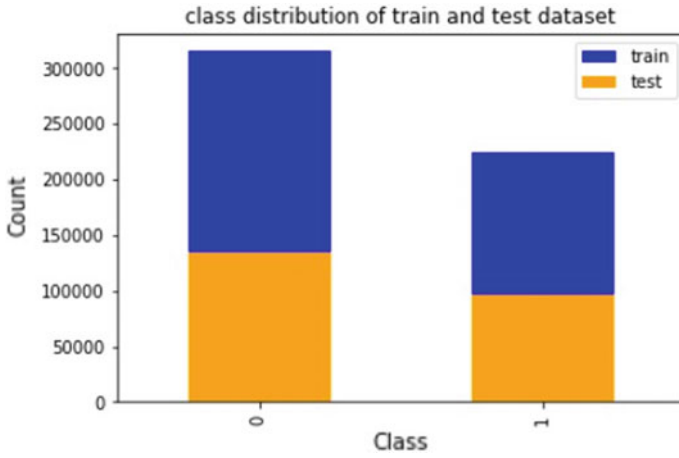


Fig. 1 Class distribution of test and train dataset

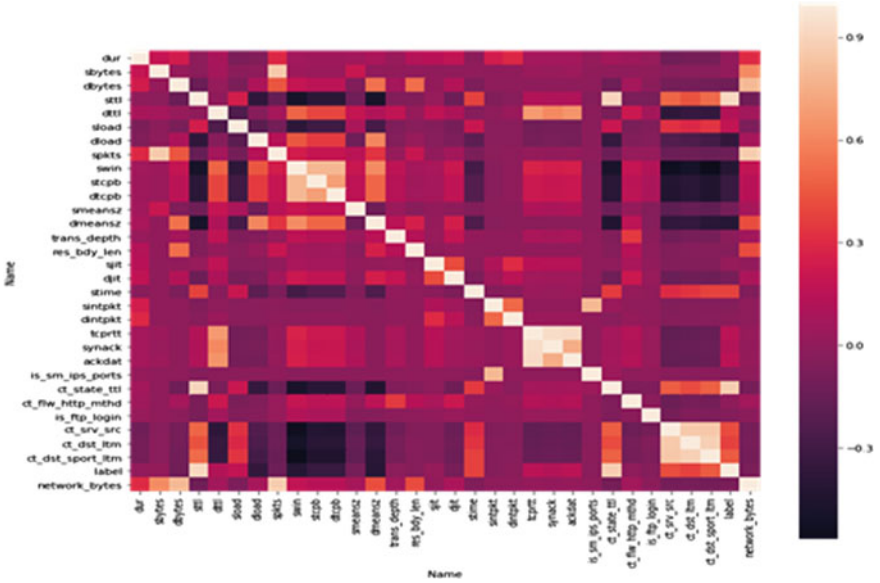


Fig. 2 Heat map for correlation matrix

about overfitting during execution, the quantity of normal records is decreased to a comparable number of attack records. The quantity of normal records is yet kept higher to lessen false positive rates. The dataset is then separated into preparing and testing datasets with the end goal that the preparation dataset possesses 70% of the complete information.

2. **Data Cleaning:** UNSW-NB15 is an enormous dataset and subsequently contains a scope of irregularities too. The information contains missing values and wrong values which are supplanted by the mode worth of these very features. Fixing parallel sections, for example, “is_ftp_login” which are in a scope of qualities is likewise in this manner rectified to 1 and 0 which permits in more speculation of each grouping while at the same time permitting a level of deviation to gain.
3. **Data Transformation:** Data transformation is viewed as a fundamental advance in data preprocessing as it guarantees most extreme information quality and furthermore helps in better examination of the model, and it accordingly incorporates different sub-steps.
 - (a) **Log Transformation:** As the numerical values in the dataset contain values close to zero just as numbers bigger than 0, log change on the dataset can react to the skewness of the data as qualities near nothing and bigger than zero can mutilate the dataset. In this way, we use \log_{1p} on the dataset to deliver just positive values and produce a homogenous dataset [12].
 - (b) **Normalization:** Normalization is a scaling procedure to change the information when the data sources have various ranges. We find out the mean and standard deviation of the features from which we then, at that point, subtract the input alongside the mean. These outcomes in a set of values with the mean 0 and standard deviation are 1. This considers better classification and paces up execution too. Figure 2 represents the correlation matrix.
 - (c) **Encoding the Labels:** This process is used to convert the features “proto”, “service”, “state”, and “attack_cat” which are present in categorical values into numerical values ranging from zero to one less than the number of classes. Thus, this converts the data into machine-readable language.
4. **Data Reduction:** As the UNSW-NB15 dataset has 47 highlights of which all have diverse significance, it is critical to eliminate the highlights which do not add to arrangement as much as different highlights to eliminate the opportunity of overfitting highlights of less significance and underfitting highlights of more significance. A high number of features likewise bring about dimensionality issues while model preparing because of the great complexity it faces.
 - (a) **Correlation Matrix:** Features with high correlation will have a similar impact on the output of the model. Subsequently, when two features have a high relationship, we can drop one of them. The heat map is produced giving the output, and a feature is dropped if its correlation surpasses 0.90. The columns dropped are “sloss”, “dloss”, “dpkts”, “dwin”, “ltime”, “ct_ftp_cmd”, “ct_srv_dst”, “ct_src_ltm”, “ct_src_dport_ltm”, and “ct_dst_src_ltm”.

Fig. 3 Representation of evaluation metrics

Actual →	Negative	Positive
Predicted ↓		
Negative	True Negative (TN)	False Positive (FP)
Positive	False Negative (FN)	True Positive (TP)

- (b) Extra Trees Classifier: To additionally tackle this, the extra trees classifier was carried out as it is an ensemble learning technique that helps in unsupervised learning too. The extra trees classifier likewise gives the level of significance of each feature which considers manual threshold entry based on the outcomes.

The classifier was run on entropy criteria with its “n_max features” set as 2 and yielded a graph.

As per the graph, the threshold was set to 0.01, and thus, the top 24 features were selected for the implementation. These are the final features selected:

“sttl”, “dttl”, “swin”, “stime”, “tcprrt”, “ct_state_ttl”, “ct_srv_src”, “ct_dst_ltm”, “ct_dst_sport_ltm”, “sbytes_log1p”, “dbytes_log1p”, “sload_log1p”, “dload_log1p”, “spkts_log1p”, “stcpb_log1p”, “dtcpb_log1p”, “smeansz_log1p”, “dmeansz_log1p”, “sjit_log1p”, “djit_log1p”, “network_bytes_log1p”, “state_log1p”, “service_log1p”, “proto_log1p”

4 Evaluation Metrics

The various models implemented and compared are done by their measure of performance on some commonly used metrics—accuracy, precision, recall, and F-score. These together form the confusion matrix which helps in selecting the best implementation as well as insight into how to improve the implemented model. These metrics work mainly on four values produced by the results—true positive, true negative, false positive, and false negative (Fig. 3).

Calculation of precision can be done using the confusion matrix as follows:

$$P = \frac{TP}{TP + FP}$$

where TP is true positive, FP is false positive, TN is true negative, and FN is false negative.

To calculate accuracy, confusion matrix is as follows:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}}$$

To calculate the recall (R), confusion matrix is as follows:

$$\text{Recall } (R) = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

To calculate the F -measure (F), confusion matrix is as follows:

$$F = \frac{2 * P * R}{P + R}$$

While dealing with multiclass classification, these metrics work differently too as true positive, true negative, false positive, and false negative values are derived from different classes. The three commonly used types are micro, macro, and weighted.

5 Proposed Methodology

For binary:

Since classification in binary does not need high complexity, the complexities taken for preparing were not huge to stay away from overfitting. In any case, to avoid underfitting the information, more than one model was carried out with the use of Adam analyzer and compared about for every one of the CNN and RNN models.

Convolutional neural network:

Parameters	Accuracy (%)	Precision (%)	Recall (%)	F -score (%)
109,217	98.93	97.80	99.41	98.70
285473	98.89	97.77	99.69	98.70

Recurrent neural network:

Parameters	Accuracy (%)	Precision (%)	Recall (%)	F -score (%)
49,579	98.86	97.53	99.78	98.64
102,849	98.88	97.67	99.70	98.67

For multiclass:

Since multiclass issues require higher complexity because of the concentrated idea of their classification, another model with higher boundaries was acquainted to keep away from the conceivable event of underfitting and in this manner better examination.

Convolutional neural network:

Parameters	Accuracy (%)	Precision macro (%)	Precision weighted (%)	Recall macro (%)	Recall weighted (%)	F-score macro (%)	F-score weighted (%)
109,370	93.89	61.77	94.19	49.88	93.89	50.10	93.47
285,626	93.95	64.74	94.51	51.94	93.95	51.19	93.97
565,498	94.20	68.68	94.94	50.98	94.20	49.55	93.46

Recurrent neural network:

Parameters	Accuracy (%)	Precision macro (%)	Precision weighted (%)	Recall macro (%)	Recall weighted (%)	F-score macro (%)	F-score weighted (%)
50,858	93.89	46.53	93.13	46.25	93.89	44.22	93.06
103,146	94.09	67.73	93.77	47.47	94.09	48.17	93.24
148,394	93.74	49.96	94.47	51.92	93.74	50.17	93.92

As per the results, the proposed method for multiclass classification is CNN with Adam optimizer and categorical cross-entropy for the loss. The model runs for ten epochs with 32 sized mini batches.

It is also seen that this implementation gave the best overall results in the multiclass approaches yet. Next, the testing set failures as they allow a deeper insight to overfitting of attacks as seen previously are analyzed which do not show a bias to any attack group suggesting the minimizing of any dimensionality problem. This is hence the final proposed implementation of the multiclass classification in the UNSW-NB15 dataset.

6 Conclusion

Various models for detection of network attacks are compared in this paper. The proposed model for binary classification is convolutional neural network (CNN) from the two being compared being—CNN and RNN. The implementation reached an accuracy of 98.93% with its precision as 97.08%, recall as 99.41%, and *f*-score as 98.70%, highlighting the efficiency of convolutions and recurrence working well together. Multiclass involved the concatenation of certain attacks for its maximum macro-averages due to the unbalanced nature of the data. This implementation was best implemented with convolutional neural network (CNN) which reached an accuracy of 94.20%. This paper gives an understanding to how the two widely used architectures—CNN and RNN—work on the UNSW-NB15 dataset.

In paper [13] the experimental study of proposed method involved large datasets of about 75,000 samples with more than two-thirds consisting of malware samples and benign samples forming the rest. By performing similarity mining of the innumerable obfuscations of extended x86 IA-32 (opcodes) found in these malware samples, we

were successfully able to detect and classify unknown malware that had escaped from traditional detection methods. In paper [14], the existing benchmark datasets are not representing the comprehensive representation of the modern orientation of network traffic and attack scenarios. Paper [15], proposes an effective IDS by using hybrid data optimization which consists of two parts: data sampling and feature selection, called DO_IDS. In data sampling, the Isolation Forest (iForest) is used to eliminate outliers, genetic algorithm (GA) to optimize the sampling ratio, and the Random Forest (RF) classifier as the evaluation criteria to obtain the optimal training dataset. Paper [16–18] suggest software-defined networking (SDN), increasingly replacing conventional networking especially in the IoT, limits the features that can be used to detect botnets. Paper [19–21] give an insight on usage of Deep Neural Networks in cloud computing and IoT environment.

References

1. Mukherjee B, Heberlein LT, Levitt KN (1994) Network intrusion detection. *IEEE Netw* 8(3):26–41
2. Mishra P, Varadharajan V, Tupakula U, Pilli ES (2018) A detailed investigation and analysis of using machine learning techniques for intrusion detection. *IEEE Commun Surv Tutor*
3. Staudemeyer RC (2015) Applying long short-term memory recurrent neural networks to intrusion detection. *S Afr Comput J* 56(1):136–154
4. Moustafa N, Slay J (2015) The significant features of the UNSW-NB15 and the KDD99 data sets for network intrusion detection systems. In: 2015 4th international workshop on building analysis datasets and gathering experience returns for security, Nov 2015
5. Sharma P, Sengupta J, Suri PK (2019) Survey of intrusion detection techniques and architectures in cloud computing. *Int J High Perform Comput Netw* 13(2):184–198
6. Kamarudin MH, Maple C, Watson T (2019) Hybrid feature selection technique for intrusion detection system. *Int J High Perform Comput Netw* 13(2):232–240
7. Kumar P, Kumar R, Gupta GP, Tripathi R (2021) A distributed framework for detecting DDoS attacks in smart contract-based blockchain-IoT systems by leveraging fog computing. *Trans Emerg Telecommun Technol* 32(6):e4112
8. Sarker IH, Abushark YB, Alsolami F, Khan AI (2020) IntruDTree: a machine learning based cyber security intrusion detection model. *Symmetry* 2020
9. Kanimozhi V, Jacob P (2019) UNSW-NB15 dataset feature selection and network intrusion detection using deep learning. *Int J Recent Technol Eng (IJRTE)*
10. Kocher G, Kumar G (2021) Analysis of machine learning algorithms with feature selection for intrusion detection using UNSW-NB15 dataset. *Int J Netw Secur Appl (IJNSA)*
11. Liu H, Lang B (2019) Machine learning and deep learning methods for intrusion detection systems: a survey. State Key Laboratory of Software Development Environment, Beihang University, Beijing, 11–17 Oct 2019
12. Srinivasan S, Anivilla S, Ravi V, Kp S (2020) DCNN-IDS: deep convolutional neural network based intrusion detection system. In: TechRxiv, 18–21 Oct 2020. *IEEE*
13. Venkatraman S, Alazab M (2018) Use of data visualisation for zero-day malware detection. *Secur Commun Netw* 2018:13 pages. Article ID 1728303. <https://doi.org/10.1155/2018/1728303>
14. Moustafa N, Slay J (2015) UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In: Military communications and information systems conference (MilCIS), Canberra

15. Ren J, Guo J, Qian W, Yuan H, Hao X, Jingjing H (2019) Building an effective intrusion detection system by using hybrid data optimization based on machine learning algorithms. *Secur Commun Netw*
16. Letteri I, Penna GD, Gasperis GD (2019) Security in the internet of things: botnet detection in software-defined networks by deep learning techniques. *Int J High Perform Comput Netw* 15(3–4):170–182
17. Zheng H, He J, Zhang Y, Wu J, Ji Z (2019) A mathematical model for intimacy-based security protection in social network without violation of privacy. *Int J High Perform Comput Netw* 15(3–4):121–132
18. Kumar P, Gupta GP, Tripathi R (2021) Toward design of an intelligent cyber attack detection system using hybrid feature reduced approach for IoT networks. *Arab J Sci Eng* 46(4):3749–3778
19. Kumar R, Kumar P, Tripathi R, Gupta GP, Gadekallu TR, Srivastava G (2021) Sp2f: a secured privacy-preserving framework for smart agricultural unmanned aerial vehicles. *Comput Netw* 187:107819
20. Nagisetty A, Gupta GP (2019) Framework for detection of malicious activities in IoT networks using keras deep learning library. In: 2019 3rd international conference on computing methodologies and communication (ICCMC), Mar 2019. IEEE, pp 633–637
21. Kumar P, Gupta GP, Tripathi R (2021) An ensemble learning and fog-cloud architecture-driven cyber-attack detection framework for IoMT networks. *Comput Commun* 166:110–124

Toward Big Data Various Challenges and Trending Applications



Bina Kotiyal and Heman Pathak

Abstract With the continuous growth in the data and its use as a resource for analytic knowledge continues to help companies develop, the need for innovative approaches, tools, and strategies to extract actionable insights is becoming increasingly important. The data collected from myriad sources like social media, search engines, and the Internet of Things has developed substantial opportunities concerning the business to business industrial organizations. Big data (BD) computing is classified as batch and stream computing based on the processing types. Batch computing is performed when data is at rest, whereas real-time computing is performed when data is in motion. In the present era, real-time stream processing is in demand as the massive data generated has to be handled speedily to meet the business or organization requirements. BD analytics is used to get the big insight from this data. However, cleansing, interpreting, and analyzing such massive databases present hurdles in marketing, particularly in terms of making real-time decisions. This paper throws light on the various issues and challenges associated to BD. Most of the challenges are associated to the preprocessing phase of BD. It also presents the diverse applications of BD.

Keywords Big data · Big data analytics · Batch processing · Real-time processing

1 Introduction

In the present era, the data is generated by various sources such as social networking sites, sensors, and mobile devices. The data contains different data types. High dimension, heterogeneous, unstructured, incomplete, noisy, missing data, low quality, etc., are the unique features of BD which also changes the statistical and data analysis approaches. The various social networking Websites such as Yahoo, Face-

B. Kotiyal (✉) · H. Pathak
Gurukul Kangri Vishwavidyalaya, 60, Rajpur Road, Dehradun, Uttarakhand, India
e-mail: kotiyalbina@gmail.com

H. Pathak
e-mail: hpathak@gkv.ac.in

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
D. P. Agrawal et al. (eds.), *Cyber Security, Privacy and Networking*, Lecture Notes in Networks and Systems 370, https://doi.org/10.1007/978-981-16-8664-1_20

219

book, and Google originated the term to analyze hefty amounts of data. BD leads to BD; however, this huge amount of data is not meaningful because it contains noisy or abnormal data according to the author [1]. A lot of research has been done in the different areas of BD, but still this field needs to be more explored. BD leads to various types of issues in BD analytics such as storage, fault tolerance, quality of data, and security aspect. The data will be of low quality [2]. The performance of a system will be reduced even though we use advanced algorithms due to the poor data representations [3]. The different types of analytics are text analytics, audio analytics, video analytics, and social media analytics. The user opinions are extracted and analyzed for the various business purposes for decision making by understanding the behavior of the user. Our contribution in this paper:

1. We have performed a systematic literature survey in this area.
2. We have identified the various issues and challenges pertaining to BD.
3. We have thrown light on the applications of BD.

The paper is organized as follows: Sect. 2 discusses the processing varieties of BD, Sect. 3 is focused on the challenges, Sect. 4 presents the related work, Sect. 5 discusses the various applications, and at last Sect. 6 presents the conclusion.

2 Big Data Processing Varieties

With the continuous growth in the data, and its use as a resource for analytic knowledge continues to help companies develop, the need for innovative approaches, tools, and strategies to extract actionable insights is becoming increasingly important. The researchers have given two patterns in this field: batch processing and real-time processing.

Batch Processing. The collection of structured and unstructured datasets for developing a future strategy from the machine learning standpoint is known as batch processing. The data generated and collected through various mediums such as search engine searches, click behavior, social media, sensors, locations, smart devices, transactions, and sales information forms the data for batch processing. In batch processing, the large amount of data is processed at once. In this type of processing, latency is not a concern rather than throughput. Hadoop is introduced as a solution for processing this massive data generated or captured through various mediums [4, 5].

Real-Time Processing. Real-time processing processes the current datasets. It works on the continuous stream of inputs and generates output without delay. It handles dynamic, constantly changing, and multi-format data types within a real-time milieu. However, more efficient and fast approaches are needed to analyze this type of data. The objective of handling in real time is to analyze and process the data with minimum latency. Apache spark is used for real-time processing. It overcomes the problem of Hadoop [6].

3 Big Data Challenges

BD brings many challenges and issues with its characteristics. The following mentioned are the challenges that are needed to be solved:

Heterogeneity and Incompleteness. The real-world data generated from different sources are heterogeneous in form and incomplete [7]. Therefore, the continuously increasing size of BD and extracting value from it are the challenges faced. The layout, diversity, and organization of the streaming data must all be reflected in an effective data presentation.

Dimension Reduction. BD dimension reduction is the problem associated with the huge data also known as ‘Curse of Dimensionality’ which has many features which is directly related to vast stores and increasing the scale up of BD systems [8].

Imperfect Data. Imperfect data deals with the techniques associated with missing value or noise. Missing values can result in bad decision making and can be a potential reason for the loss of efficiency in the extraction of knowledge; therefore, it is very necessary to deal with noisy data [9]. The data often contains the noise in it which can affect the input or output or both of them and thus leads to the poor quality of data.

Class Imbalance. Imbalanced data or class imbalance is the challenge associated with large growing data also known as the volume of data and assuming that the data are not correctly classified across the distribution [10].

Unstructured Data. The unstructured data constitute a major portion of the data. Also with extraordinary expansion in the extent of data, researchers face a huge issue in handling unstructured data [11].

Feature Engineering: This problem is closely related to high dimensionality. In this kind of learning domain, knowledge is used to generate the features. But finding the highly relevant feature is the time-consuming process in preprocessing [12].

Uncertainty. Uncertainty consists of imperfect or unknown information. This challenge is associated with the veracity of data. It is present in all phases of BD learning. The various types of uncertainty in the data lead to less effective and accurate results [13].

Scalability: In massive data analysis, scalability is a major concern. Existing resources cannot keep up with the data because it is rising at an exponential rate. To handle the exponential growth of data scalable frameworks, algorithm needs to be developed by the researchers [14].

Integration. The uniform view of data collected through different mediums is known as integration [15]. This data can be in any format. A strategy for data integration should be developed to help relatively effective performance across several datasets.

Data Quality: The generation of data is done through different mediums. Data quality problem is associated with the inconsistent data generation which leads to poor data. Generating high-quality data from the data consistency is another challenge in BD [16].

Timeliness: The time-sensitive process has time as a prime factor such as for alleviating the security threats, and preventing fraud. Scalable architectures or platforms are required to enable continuous processing of streaming data that can be implemented to develop data timeliness [17]. The key problem is always to build a distributed architecture which can combine local representation of data into a global perspective with the least amount of latency possible between communication nodes.

Security. BD cannot be handled by one single system; therefore, *big repositories* are required to store this huge data. This makes it vulnerable to criminals or people targeting to the confidentiality of the information. Therefore, security will always be a major concern to BD [18].

Privacy: The most difficult aspect of large data stream analytics is coming up with strategies to preserve data before it is analyzed [19]. International Data Cooperation (IDC) revealed that the information is not effectively protected which needs guard, and this has created a menace to individual privacy while giving them opportunities for analyzing massive amounts of data in real time.

Visualization. The unprecedented growth in BD has generated various challenges to the existing technologies. The visualization of data in the form of graphs or through pictorial representation is becoming challenging with the enormous data generating with high speed and is the indifferent form [20].

4 Related Work

In the specific field, a lot of relevant work has been done. Some of the work is as described under:

The deficiency of efficient preprocessing techniques leads to a poor model. The inclusion of distortion in the dataset is the most prominent problem and thus making the performance of the system deteriorating. To extract the valuable information from the data, [21] has introduced smart data which can be constructively used by different organizations for intelligent decision making. The performances of machine learning algorithms are based on every preprocessing step [22].

The complexity of the data increases due to errors [23]. Preprocessing is a time-consuming process, and 80% of the time is spent on this process known as data preparation. The existing tools or techniques are manual, time-consuming and costly. Therefore, this area needs to be focused on.

According to the author [24], size is only one dimension of BD out of many dimensions. It is very important to consider the speed to which the data is arriving and the variety. The paper focuses on the natures of BD like audio, video, social

media, and text. Other papers focus on structured data using predictive analytics, whereas they do not focus on unstructured data that constitutes 95% of the BD.

As per the study of [19], the traditional data mining algorithms can be used to generate meaningful information in big data analytics, but the other researchers have not conducted many studies on this, and therefore, it is an important issue.

The author has measured the area of BD and examines the various kind of techniques employed for processing and analytics [25]. The author has analyzed the genesis of BD through structuralism and functionalism. It has discussed the various management tools based on batch processing and stream processing. It has also discussed the open research challenges in BD. The current tool and technologies are not capable of solving the data problems entirely for processing BD and analytics. Therefore, more research needs to be carried out in the sub-areas of BD. It also discusses some emerging technologies for BD that can overcome the existing problem of BD.

A systematic literature survey is performed by [26] and presented challenges faced by the organization and also discussed the analytical methods adopted by the organization to overcome them. They have given a conceptual classification of BD challenges such as data challenges (related to the characteristics), process challenges (related to how technique), and management challenges (related to ethical aspects).

The various definitions of BD given by other authors are also discussed [17]. BD is described by the author as a massive amount of data that differs from ordinary data in terms of size and format. It is a combination of unstructured, semi-structured, and structured data. It has covered the major concerns and challenges associated with BD, as well as the solutions' limitations. Hadoop MapReduce for batch-based data processing and Apache Spark for real-time data methods are adopted to reduce the latency in processing.

The data collected from myriad sources like social media, search engines, and the Internet of Things has developed substantial opportunities concerning the business-to-business industrial organizations. However, cleansing, interpreting, and analyzing such massive databases presents hurdles in marketing, particularly in terms of making real-time decisions [27].

BD computing is classified as batch and stream computing based on the processing types [28]. Batch computing is performed when data is at rest, whereas real-time computing is performed when data is in motion. In the present era, real-time stream processing is in demand as the massive data generated has to be handled speedily to meet the business or organization requirements.

Author [11] has performed research about the limitations of the existing information extraction techniques. The various problems related to unstructured data are explored independently. The preprocessing phase needs to be focused to get more accurate results.

Table 1 shows the literature survey performed on the BD in a tabular format with limitations and future research directions.

Table 1 Summary of literature survey

Author	Focuses	Limitations	Future work
[24]	Presented various definitions of BD To gain valid and valuable insights from BD On analytics pertaining to unstructured data (constitute 95% of BD)	Only consider predictive analytics which is used for structured data	The emergence of novel technologies will lead to the field of real-time analysis due to growth in location-aware social media and mobile app
[29]	Traditional mining algorithms are not competent in handling BD The KDD process is used as a framework and is summarized into three parts: input, analysis, and output Discuss some open issues that we may face in BD analytics	Not given a solution to the issues	To predict the behavior of the user How to protect data To apply the traditional algorithm for BD mining
[26]	Performed SLR Discussed the various challenges Presented the analytics methods employed to handle the challenges	It does not discuss any tool	Nil
[17]	Various definition of BD Discussed the issues and challenges A brief comparison of Hadoop MapReduce and Apache Spark	Only theoretical aspects are given	More technologies could be employed
[21]	Discussion on preprocessing techniques Two classifiers adapted for generating smart data	Just the classification algorithms are used	Multiclass and imbalanced problem

(continued)

Table 1 (continued)

Author	Focuses	Limitations	Future work
[27]	The structure and unstructured datasets are linked to batch versus real-time handling Much work is done on structured datasets with batch processing For B2B industrial marketing, the actual processing of unorganized analysis is needed	Paper is conceptual Poor decision making leads to poor targeting and increasing latency time	Practical implementation is missing Make a strategy on the limitations of the decision-making process
[28]	Focuses on large analytics in real time	Protecting the actual time data	Social media preprocessing
[30]	Focuses on handling unstructured data	Real-time processing of data is restricted	Use of BD analytics to the educational sector, geo-science, etc.
[13]	Focuses on uncertainty AI yields more accurate results than other techniques	Uncertainty related to the dataset not discussed. Only analytics uncertainty is focused	Efficiently model uncertainty in ML and NLP
[11]	Focuses on unstructured and multidimensional BD Various types of information extraction via text, audio, etc. Independent challenges identified based on unstructured data	Usability improvement model not given for handling the unstructured data	Increasing the efficiency of the system via preprocessing of a variety of data

5 Applications Using Big Data

The term “BD” refers to large amounts of information generated both internally and externally. This data helps in great decision making when handled gently. Some of the prominent applications are as follows:

Bank Sector. BD is used in banks and helps in providing better services in the bank such that collection of cash and financial management. It has created profit for the banks and also diminished customers’ stress. It helps get the high demand for new branches.

Tourism: In the present era rather than agencies tourists would like to go through the digital world [31]. It helps collect information about the tourists, the tourist places, etc. The airlines can deliver effective services when they have complete information about passengers. Beneficial offers can be given to customers based on the idea of geo-location, weather, and traffic.

Disaster Management: The development of new technologies helps the meteorologists to forecast weather conditions more accurately. It can evaluate factors such as water level, wind pressure, and temperature to find the possibility of disaster. Disasters like earthquakes can be monitored and timely people can be warned by disaster management specialists.

Agriculture. BD is used in the field of Agriculture. It uses historical data, machine-generated data, and real-time data to address real-world problems. Agricultural IoT creates a vast volume of agricultural data, which can be employed to automate the firm's watering system, allowing farmers to move on to more important matters [32].

Cloud Computing: Nowadays, these two technologies are blended for providing more innovative solutions for the present problem [33]. Cloud gives us various services and thus gives us a tool to perform these services. One such service is the extraction of required data. The cloud server is used to store the data, and relational information is extracted employing BD techniques.

Education: BD in education sector can yield unique results that can provide new data-driven approaches for teaching students as it is the mainstay of any nation. It includes the records of students and could store, manage, analyze the massive datasets [34].

Customer-Oriented Service: It is imperative to understand the need of the customer as they are the assets of any business. Understanding the need for customer and customer satisfaction is the key to have a successful business. BD tells us what our customers are looking for and gives us the best results. It helps us in identifying the requirement of customers [35].

Social Media Sector: It is among the most widespread sectors in the present world. Here, social media is used for knowing the opinion of people, and this can be used for various purposes. One such purpose can be providing opportunities to the digital market in identifying their customers using AI. It gives a better idea of the customer and thus helps in decision making [36].

Cyber-Physical Systems: Computer security networks are used to guard the sensitive information of the organization and government. BD is a strategy for gathering, organizing, and storing information. Technological innovation is often used to properly defend data. BDA is significant in overcoming serious security and privacy challenges, as well as enabling numerous businesses to access data and obtain a thorough understanding of business [37].

Healthcare. The healthcare industry is data-intensive [4]. It entails techniques for evaluating huge quantities of electronic data related to the patient medical care and well-being. Traditional software and hardware have a hard time measuring this data since it is so diverse.

6 Conclusion

The research done in this paper concludes that the deficiency of efficient preprocessing techniques leads to a poor model. The inclusion of distortion in the dataset is the most prominent problem and thus deteriorates the performance. To extract the valuable information from the data, smart data is constructively used by different organizations for intelligent decision making. The performances of machine learning algorithms are based on every preprocessing step. However, it is a time-consuming process and 80% of the time is spent on this process known as data preparation. The paper has discussed the myriad issues, challenges, and applications of BD which need to be focused so that the results can be enhanced. Some papers have given focus on techniques that are manual, time-consuming, or costly. Therefore, it requires more research into this field. In the future the tools for BD will be introduced.

References

1. Tsai CW, Lai CF, Chao HC, Vasilakos AV (2015) Big data analytics: a survey. *J Big Data* 2:1–32. <https://doi.org/10.1186/s40537-015-0030-3>
2. Al-Taie MZ, Kadry S, Lucas JP (2019) Online data preprocessing: a case study approach. *Int J Electr Comput Eng* 9:2620–2626. <https://doi.org/10.11591/ijece.v9i4.pp2620-2626>
3. Najafabadi MM, Villanustre F, Khoshgoftaar TM, Seliya N, Wald R, Muharemagic E (2015) Deep learning applications and challenges in big data analytics. *J Big Data* 2:1–21. <https://doi.org/10.1186/s40537-014-0007-7>
4. Galetsi P, Katsaliaki K, Kumar S (2020) Big data analytics in health sector: theoretical framework, techniques and prospects. *Int J Inf Manage* 50:206–216. <https://doi.org/10.1016/j.ijinfomgt.2019.05.003>
5. Kotiyal B, Kumar A, Pant B, Goudar RH (2013) Big data: mining of log file through Hadoop. In: *Proceedings of the international conference on human-computer interaction, ICHCI 2013*, pp 1–7. <https://doi.org/10.1109/ICHCI-IEEE.2013.6887797>
6. Mohamed A, Najafabadi MK, Wah YB, Zaman EAK, Maskat R (2020) *The state of the art and taxonomy of big data analytics: view from new big data framework*. Springer, Netherlands. <https://doi.org/10.1007/s10462-019-09685-9>
7. Guan Z, Ji T, Qian X, Ma Y, Hong X (2017) A survey on big data pre-processing. In: *5th international conference on applied computing and information technology*, pp 241–247. <https://doi.org/10.1109/ACIT-CSII-BCD.2017.49>
8. ur Rehman MH, Liew CS, Abbas A, Jayaraman PP, Wah TY, Khan SU (2016) Big data reduction methods: a survey. *Data Sci Eng* 1:265–284. <https://doi.org/10.1007/s41019-016-0022-0>
9. Ezzine I, Benhlima L (2018) A study of handling missing data methods for big data. In: *Colloquium in information science and technology CIST*, Oct 2018, pp 498–501. <https://doi.org/10.1109/CIST.2018.8596389>
10. Fernández A, del Río S, Chawla NV, Herrera F (2017) An insight into imbalanced big data classification: outcomes and challenges. *Complex Intell Syst* 3:105–120. <https://doi.org/10.1007/s40747-017-0037-9>
11. Adnan K, Akbar R (2019) An analytical study of information extraction from unstructured and multidimensional big data. *J Big Data* 6:1–38. <https://doi.org/10.1186/s40537-019-0254-8>
12. L'Heureux A, Grolinger K, Elyamany HF, Capretz MAM (2017) Machine learning with big data: challenges and approaches. *IEEE Access* 5:7776–7797. <https://doi.org/10.1109/ACCESS.2017.2696365>

13. Hariri RH, Fredericks EM, Bowers KM (2019) Uncertainty in big data analytics: survey, opportunities, and challenges. *J Big Data* 6:1–16. <https://doi.org/10.1186/s40537-019-0206-3>
14. Acharjya DP, Ahmed K (2016) A survey on big data analytics: challenges, open research issues and tools. *Int J Adv Comput Sci Appl* 7:511–518. <https://doi.org/10.14569/ijacsa.2016.070267>
15. Marjani M, Nasaruddin F, Gani A, Karim A, Hashem IAT, Siddiqa A, Yaqoob I (2017) Big IoT data analytics: architecture, opportunities, and open research challenges. *IEEE Access* 5:5247–5261. <https://doi.org/10.1109/ACCESS.2017.2689040>
16. Valenzuela-Escárcega MA, Hahn-Powell G, Hicks T, Surdeanu M (2015) A domain-independent rule-based framework for event extraction. In: *ACL-IJCNLP 2015—53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing system demonstrations proceeding*, pp 127–132. <https://doi.org/10.3115/v1/p15-4022>
17. Wani MA, Jabin S (2018) Big data: issues, challenges, and techniques in business intelligence. In: *Advances in intelligent systems and computing*. Springer Verlag, pp 613–628. https://doi.org/10.1007/978-981-10-6620-7_59
18. Maqbool Q, Habib A (2019) Big data challenges. *Control Eng* 66:33. <https://doi.org/10.4172/2324-9307.1000133>
19. Taleb I, Serhani MA, Dssouli R (2018) Big data quality: a survey. In: *IEEE international congress on big data. BigData congress 2018—part of the 2018 IEEE world congress on services*. IEEE, pp 166–173. <https://doi.org/10.1109/BigDataCongress.2018.00029>
20. Chahal H, Jyoti J, Wirtz J (2018) Understanding the role of business analytics: some applications. Springer, Singapore. <https://doi.org/10.1007/978-981-13-1334-9>
21. García-Gil D, Luengo J, García S, Herrera F (2019) Enabling smart data: noise filtering in big data classification. *Inf Sci (Ny)* 479:135–152. <https://doi.org/10.1016/j.ins.2018.12.002>
22. Alam S, Yao N (2019) The impact of preprocessing steps on the accuracy of machine learning algorithms in sentiment analysis. *Comput Math Organ Theory* 25:319–335. <https://doi.org/10.1007/s10588-018-9266-8>
23. Keswani B (2018) Enhanced approach to attain competent big data pre-processing. In: *4th international conference on cyber security*, pp 524–527
24. Gandomi A, Haider M (2015) Beyond the hype: big data concepts, methods, and analytics. *Int J Inf Manage* 35:137–144. <https://doi.org/10.1016/j.ijinfomgt.2014.10.007>
25. Yaqoob I, Hashem IAT, Gani A, Mokhtar S, Ahmed E, Anuar NB, Vasilakos AV (2016) Big data: from beginning to future. *Int J Inf Manage* 36:1231–1247. <https://doi.org/10.1016/j.ijinfomgt.2016.07.009>
26. Sivarajah U, Kamal MM, Irani Z, Weerakkody V (2017) Critical analysis of big data challenges and analytical methods. *J Bus Res* 70:263–286. <https://doi.org/10.1016/j.jbusres.2016.08.001>
27. Jabbar A, Akhtar P, Dani S (2019) Real-time big data processing for instantaneous marketing decisions: a problematization approach. *Ind Mark Manag* 1. <https://doi.org/10.1016/j.indmarman.2019.09.001>
28. Kolajo T, Daramola O, Adebisi A (2019) Big data stream analysis: a systematic literature review. *J Big Data* 6:1–30. <https://doi.org/10.1186/s40537-019-0210-7>
29. Krawczyk B (2016) Learning from imbalanced data: open challenges and future directions. *Prog Artif Intell* 5:221–232. <https://doi.org/10.1007/s13748-016-0094-0>
30. Madaan A, Sharma V, Pahwa P, Das P, Sharma C (2018) Hadoop: solution to unstructured data handling. *Adv Intell Syst Comput* 654:47–54. https://doi.org/10.1007/978-981-10-6620-7_6
31. Amudhavel J, Padmapriya V, Gowri V, Lakshmi Priya K, PremKumar K, Thiagarajan B (2015) Perspectives, motivations and implications of big data analytics. In: *ACM international conference proceeding series*, pp 1–5. <https://doi.org/10.1145/2743065.2743099>
32. Sharma S, Rathee G, Saini H (2018) Big data analytics for crop prediction mode using optimization technique. In: *5th international conference on parallel, distributed and grid computing*. IEEE, pp 760–764. <https://doi.org/10.1109/PDGC.2018.8746001>
33. Balachandran BM, Prasad S (2017) Challenges and benefits of deploying big data analytics in the cloud for business intelligence. *Procedia Comput Sci* 1112–1122. <https://doi.org/10.1016/j.procs.2017.08.138>

34. Shatnawi MQ, Yassein MB, Abuein Q, Nsuir L (2019) Big data analytics tools and applications: survey. In: ACM international conference proceeding series, pp 1–4. <https://doi.org/10.1145/3368691.3368741>
35. Lv Z, Song H, Basanta-Val P, Steed A, Jo M (2017) Next-generation big data analytics: state of the art, challenges, and future research topics. *IEEE Trans Ind Inform* 13:1891–1899. <https://doi.org/10.1109/TII.2017.2650204>
36. Desai PV (2018) A survey on big data applications and challenges. In: Proceedings of the international conference on inventive communication and computational technologies ICICCT 2018. IEEE, pp 737–740. <https://doi.org/10.1109/ICICCT.2018.8472999>
37. Saggi MK, Jain S (2018) A survey towards an integration of big data analytics to big insights for value-creation. *Inf Process Manag* 54:758–790. <https://doi.org/10.1016/j.ipm.2018.01.010>

Convolutional Neural Network-Based Approach to Detect COVID-19 from Chest X-Ray Images



P. Pandiaraja and K. Muthumanickam

Abstract COVID-19 is a worldwide pandemic that poses serious health hazards. COVID-19's diagnostic test sensitivity is restricted owing to specimen processing abnormalities. The discussed technique might be used in clinical practice as a computer-aided diagnostics approach for COVID-19. The use of chest X-ray pictures for detection is life-saving for both patients and clinicians. Furthermore, in nations where laboratory kits for testing are unavailable, this becomes even more critical. This work aims to demonstrate the application of deep learning for high-accuracy COVID-19 identification utilizing chest X-ray images. Image-based applications have reached a pinnacle in the last five years thanks to the widespread usage of convolutional neural networks (CNNs). CNN gathers information from images by extracting features. The enormous popularity and efficacy of CNNs have sparked a new rise in interest in deep learning. The image data space is littered with CNN models. They excel in computer vision tasks like image categorization, object identification, and image recognition. This research work attempts to discuss the CNN-based approach for detecting COVID-19 from chest X-ray images.

Keywords COVID-19 · X-ray imaging · Deep learning · Convolutional neural network

1 Introduction

VIRUS became a deadly illness as many of us every location the world area unit affected and most of them area unit is dead. The virus mainly spreads through coughs

P. Pandiaraja (✉)

Department of Computer Science and Engineering, M.Kumarasamy College of Engineering, Karur, Tamil Nadu 639113, India
e-mail: sppandiaraja@gmail.com

K. Muthumanickam

Kongunadu College of Engineering and Technology, Thottiam, Tiruchirappalli, Tamil Nadu, India
e-mail: muthumanickam@kongunadu.ac.in

and sneezes of an infected person. One square measure usually merely infected by eupneic the virus if they are at intervals proximity or by having physical contact with the patient. Total of 111,740,598 cases were better-known, 86,907,998 were recovered, and one or two of 473,879 were dead therefore. There are a unit twenty 2,301,784 active cases [1]. Most of the parents infected with the Coronavirus experience sickness. There are units several tests form of a diagnostic assay, CT scan, super molecule testing, etc. for coronavirus. The foremost common checks taken for characteristic infection area unit are the agent check and super molecule [2].

An infective agent check tells that if the patient is presently infected or not; associate protein check might tell that if you had a past infection. But these tests area unit dear and can take time a prolonged time. One in all fighting with CORONA VIRUS is that the power to sight the patients early and location patients beneath with care. The target of the project is to make a picture classification model which will predict corona with a chest X-ray scan of the patients [3]. Detection this illness from radiology out taken is one of the ways that during which to make out the patients. Variety of the primary analyses displayed unique aberrations at intervals the upper-body radiograms of a sufferers plague-ridden with virus. Exhibiting virus illness presence was better-known by a board qualified specialist. Transfer education on a collection of two thousand radiograms square measure usually accustomed train four modern convolutional neural networks, at the side of ResNet18, ResNet50, Squeeze Net, and DenseNet-121, to identify CORONA illness at intervals the analyzed chest X-ray and conjointly the model most closely fits to sight COVID-19 is Res Net with high accuracy with eighteen layers deeper [4, 5]. The image of the normal person and a COVID patient.

The researches created in several location has according that the utilization of AI-based tools in resolution CT scans, based on work with X-ray, image classification problems in tending, etc. Deep learning is one in all the terribly powerful tools for psychological feature problems, learning sophisticated, and conjointly the frequency of their analysis and usage of deep education rule mistreatment the Convolutional Neural Network (CNN) that will facilitate in detection CORONA from Chest X-rays for rapid designation competently [6]. A neural network community is a form of profound neural community that is most widely recycled to analyze visible creativity in deep learning. It follows a pattern in all its work motion picture and applies identical pattern on the check data to predict the result.

The researches created in many location has according that the utilization of AI-based tools in resolution CT scans, based on coaching with X-ray pictures, image classification issues in tending, etc. [7]. Deep learning is one among the very powerful tools for psychological feature issues, learning complicated, and also the frequency of their analysis and usage of deep education rule victimization the Convolutional Neural Network (CNN) will facilitate in detection CORONA from Chest X-rays pictures for rapid abrupt.

A convolutional neural network, also known as an important category of deep neural networks, is most commonly used to analyze the visual imagination. It follows a pattern in all its coaching pictures and applies identical pattern on the check knowledge to predict the result.

1.1 Interdisciplinary

Computer vision is one of the foremost exciting divisions of technology. Heaps of analysis have been carried during this field for many years. The process of pictures becomes quicker and economical because of cloud technologies and powerful GPUs and TPUs. Cars, robots, and drones begin to grasp what we have a tendency to see in motion picture and videos. The interface “computer vision” between machines and humans can gain rather more importance inside the following few years. Computer vision is taken into account to be the most popular field within the era of AI.

It are often agitated for newbies as there square measure some challenges that the majority folks face whereas creating a transition into laptop vision. In straightforward words laptop vision may be a field of deep learning that enables the machine to spot, method pictures rather like humans [8]. In terms of parsing pictures, humans perform extraordinarily well; however, once it involves machines sleuthing objects involve multiple and sophisticated steps, together with feature extraction (edges detection, shapes), feature classification.

1.2 Library of Programming Function

OpenCV contains implementations of over 2500 algorithms! It is freely accessible for business still as tutorial functions. The library has interfaces for multiple languages, together with Python, Java, and C++.

1.3 Image Diagnosis

An image is often diagrammatic as a third-dimensional array. This is often be as a result of a machine can represent everything as numbers and in python, NumPy are often accustomed represent it whereas in C programming language it are often diagrammatic as format Mat. For images, usually, a generic word is employed known as constituents or pixel values. Within the case of color pictures, we have got three colored channels [9, 10]. Hence, colored pictures can have multiple values for single constituent values betting on the resolution and color depth; those arrays will vary in size. The color values go from zero to 255. These color channels square measure usually diagrammatic as Red inexperienced Blue (RGB).

For example, Reading pictures in OpenCV is easy; purpose to be noted here that by default, the perform reads pictures within the blue inexperienced red (BGR) format. We will browse pictures in several formats victimization further flags within the read function. The image has been properly loaded by OpenCV as a NumPy array; however, the color of every constituent has been sorted as BGR. Matplotlib lib’s plot

expects associate RGB image; thus, for an accurate show of the image, it is necessary to swap those channels.

This operation is often done either by victimization OpenCV conversion functions `cv2.cvtColor ()` or by operating directly with the NumPy array. Resizing pictures as general most laptop vision models work on mounted input shapes. True pain arises after we perform Internet scrapping to scrap image datasets. Resizing is basically useful in coaching deep learning models [11]. But totally different interpolation and down sampling functions additionally represent the umbrella of OpenCV with the subsequent parameters. Blending pictures With the magic of OpenCV, we will add or mix two pictures with the assistance of the `cv2.addWeighted()` methodology. `Add Weighted ()` perform returns NumPy array containing constituent values of the ensuing image. Mixing is nothing however the addition of 2 image matrices. thus if we wish to feature 2 pictures then which means terribly straightforward we've got to feature various 2 matrices. For aggregating 2 matrices, the scale of the 2 pictures ought to be constant.

1.4 Edge Detection

Edges in pictures square measure the points wherever brightness changes drastically and includes a variety of discontinuities like

1. Depth Discontinuities
2. Orientation Discontinuities

Edge detection has become terribly helpful for extracting options of pictures for various image recognition applications just like the classification of objects.

2 Related Works

Deep remaining associations have emerged as a gathering of unfathomably significant models showing persuading precision and respectable blend rehearses. In this paper, we inspect the multiplication definitions behind the waiting structure blocks, which suggest that the forward and in turn around signs can be clearly induced from one square to whatever other square, when using character mappings as the skip affiliations and after-development incitation. A movement of evacuation tests supports the meaning of these character mappings. This impels us to propose another excess unit, which makes planning fewer complexes and improves theory [4]. Research significance: The comprehensive type of this paper has been recognized by IEEE Internet of Things journal; if it is not all that much difficulty, allude to the journal variation. During the disease shirking and control period, our examination can be valuable in expectation, finding, and assessing for the patients spoiled with COVID-19 (the novel COVID) taking into account breathing characteristics. According to the latest

clinical investigation, the respiratory illustration of COVID-19 is not equivalent to the respiratory instances of flu and the ordinary infection. One gigantic sign that occurs in the COVID-19 is Tachypnea. People defiled with COVID-19 have all the more quick breath. Our assessment can be utilized to perceive distinctive respiratory models, and our device can be first off put to sensible use [5].

The scene of relentless discriminating respiratory disorder coronavirus 2 has brought in excess of 2.5 million occasions of coronavirus sickness (COVID-19) in the humankind up until this point, with that quantity long-lasting to create. To have power over reducing the increase of the ailment, assessing enormous amounts of assumed cases for legitimate disconnect and management is a need. Pathogenic lab examining the best level anyway is monotonous with basic fake results. Hence, elective illustrative ways are frantically expected to fight the sickness [12]. We research the effect of the convolutional network significance on its precision in the colossal degree picture affirmation setting. Our standard responsibility is a thorough evaluation of associations of extending significance using a plan with infinitesimal (3×3) convolution channels, which shows that an immense improvement for the previous workmanship arrangements can be cultivated by pushing the significance to 16–19 weight layers. These revelations were the reason of our ImageNet Challenge 2014 convenience, where our gathering got the primary and the additional locations in the localization and game plan pathways independently. We furthermore show that our depictions summarize well to other datasets, where they achieve top-tier results [13].

We are in attendance to the understanding of inauguration modules in convolutional neural associations like a center development in normal convolution and the intensity-wise distinguishable convolution movement (an intensity-wise convolution followed by a bulleted convolution). In this illumination, an intensity-wise detachable convolution can be seen as an inauguration module with a highest gigantic amount of apexes. This discernment drives us to put forward a novel significant convolutional neural association configuration stimulated by inauguration where inauguration modules have been dislocated with intensity-wise distinguishable convolutions [14]. Seven COVID are recognized to cause sickness in individuals (2, 5, and 6). A dual strain, outrageous extreme on breathing condition in COVID-19 (SARS-CoV) and Middle East respiration problem of COVID-19 (MERS-CoV), has beginnings and has associated with flare-ups of genuine breathing sicknesses in individuals (5). Though 2019-nCoV, too, is acknowledged to have a root, individual-to-singular communication has been recorded [15].

The epic (COVID-19) sickness is compacting the clinical benefits classifications across the world, and very insufficient of them is almost fading. The acknowledgment of this contamination as early as possible will help in tarnishing the spread of it as the disease is changing itself as debauched as could truly be considered typical, and as of now, there are around 4300 strains of the contamination agreeing to the reports. Experimental assessments have shown that a huge bit of the COVID-19 sufferers experience the evil impacts of a lung contamination like influenza. Thusly, it is imaginable to examine lung sickness by means of imaging procedures [16, 17].

Viruses are the most notable explanations behind respiratory tainting. The imaging disclosures of viral pneumonia are various and covers with those of other non-viral

powerful and combustible conditions. Regardless, conspicuous confirmation of the secret viral microorganisms may not for the most part be basic. There are different markers for perceiving viral microorganisms dependent on imaging plans, which are connected with the pathogenesis of viral illnesses. Diseases in a comparative virus-related intimate portion a tantamount pathogenesis of pneumonia, and the imaging plans have discernable ascribes [18]. A Standardized Dainty CNN model is adjusted from a Dainty CNN model. We acquainted a standardized level with this ideal in both preparing and examination stage. The standardized level standardizes the yield highlights, causing it to address pictures enhanced. We assess our model on LFW dataset [3]. The exactness of expression check arrives at 98.46%, which is superior to the first model.

3 Existing System Architecture

The knowledge about the project can be gained from the existing models. The COVID-19 sufferers are detected with the benefit of patients X-ray image. The main drawback is that these model takes the image as a whole and processes them to predict the result [19–23]. This can be made even more efficient by reducing all the image sizes evenly leaving the unwanted extra details in the image.

There existing system helps in knowing some of the basic and very important factors that are essential to begin the model. In this existing model, the upper-body X-ray image of in good physical person, COVID patients, bacterial pneumonia, and virus pneumonia personas are given as a contribution to the pre-trained neural network, and the output is predicted and represented in Fig. 1.

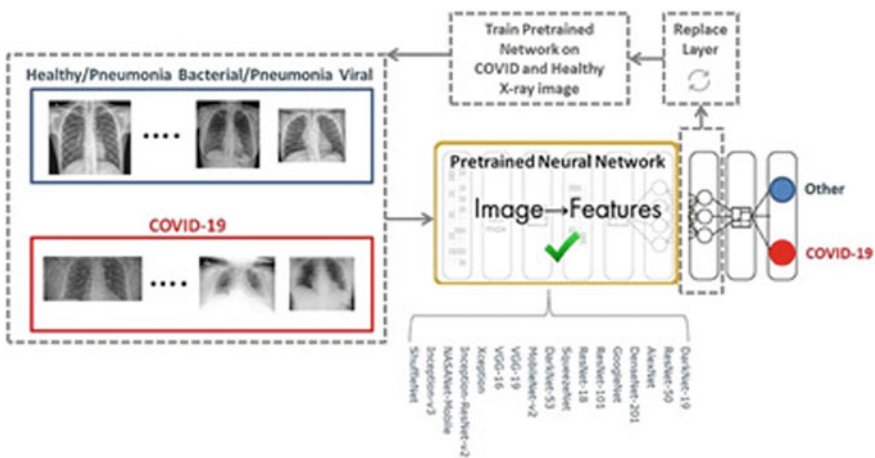


Fig. 1 Existing system architecture

There is a widespread of COVID all over the world, and all the people are affected by the deadly virus. Though the vaccines are available, the spread of virus is no way reduced. The main reason for more spread is that people are not aware whether they are affected by COVID or not. Though the COVID tests can help to identify the patients, it is taking a long time for the results and it is enough time for the virus to spread from one to other. Though lockdown, wearing mask and getting vaccinated is being followed the spread of virus exists [7, 24].

The point that can be speeded up is the duration of identifying COVID patients with the COVID test which can be reduced. Average of 14 people gets in contact with the patient for every one hour [25–29]. Reducing the duration of finding result by speed up alternate test can be finding to reduce the spread. This can be made possible with the benefit of deep learning. Convolutional network is a deep learning algorithm which is commonly used to work with images [30–36].

This can help in detecting and identifying the COVID-affected patients very quickly. All needs the upper-body X-ray image of the person. If the image of the patient upper-body X-ray is feed to the ideal, it can predict the result in no time, and thus, the patients are identified quickly, isolate, and treat the patient very quickly [37–42].

4 Proposed System Architecture

The protocol proposed is designed based on the below architecture. The model built which the base as this architecture. The additional information about the models in the architecture is discussed in this section in Fig. 2.

In this model, preprocessing and cleaning comes under image processing. If the image is very large, the time taken to predict the result will also be long. Hence, the unnecessary segments in the input can be eliminated which is done during this stage of the model. Resizing is important to make the model efficient. If the image is very large, then the processing will also take more time. Hence, the image has to be reduced by eliminating additional details/pixels in the input image [43–46].

Denoise can be used to clear the blur image.

Segmentation can be used to separate the background and the foreground.

Process the input image depending on the shapes.

4.1 Feature Engineering

The input of the model is the upper-body X-ray image of the patients. All clear images of the X-ray with average quality can be given as an input to the model. Before using the model, the model should go through training and testing through which it learns

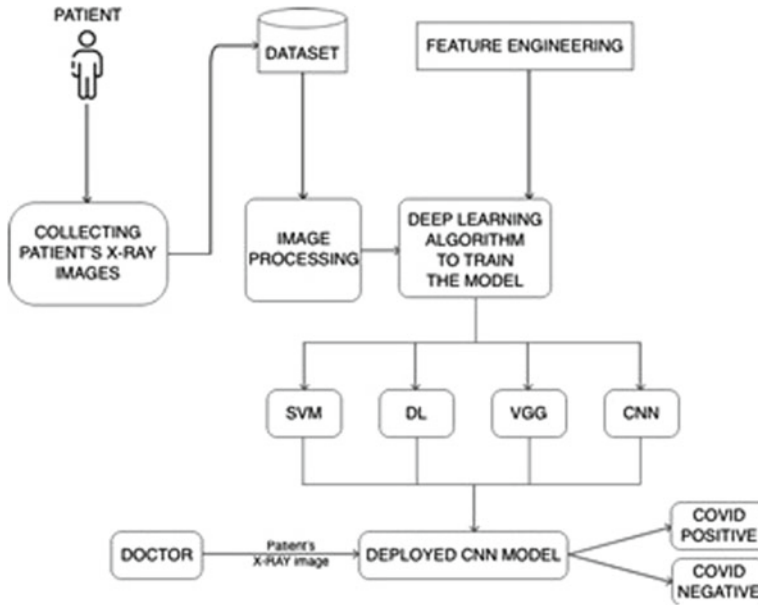


Fig. 2 Image preprocessing

the pattern in the inputs (training data) and predicts the results of testing data with help of pattern learned from the training [47–49].

Convolutional layer is the primary layer in the exemplary and is used to fetch out of the different featured from the input. In convolutional layer, the mathematical action is done in between the image and the specified size.

In common, the next layer to convolutional coating is the pooling layer. The main goal of this level is to reduce the feature map size and hence reducing the processing cost. In dense layer, all the nodes get values from all the nodes in the previous layer hence called so. Depending on the type of pooling, the pattern learned the result is made.

5 Proposed Work

The collected data has to be divided into two classes (COVID and normal). The dataset has only black and white images. Since there are only two classes possible, the final layer of the model can have a single node in Fig. 3a, b.

Find the total number of images in the dataset. Divide the dataset in the ratio 8:2 for training and testing, respectively. Create a folder named dataset with two folders nested inside that (test and train). Within test and train, create two folders named COVID and normal. Now move the images to the appropriate location using

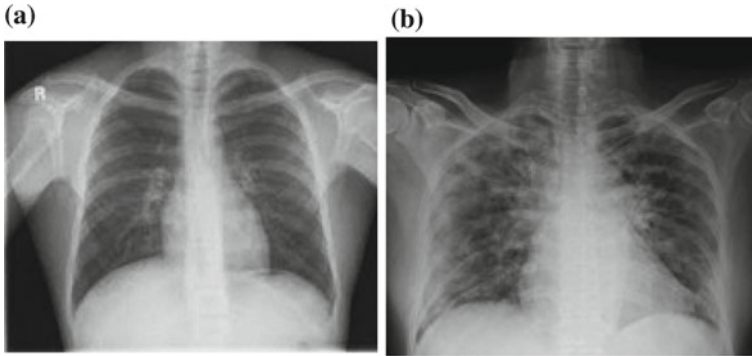


Fig. 3 a Normal patient X-ray, b COVID-19 patient X-rays

Table 1 Source library files

Key words	Use case
Numpy	Python library used for handling multi-dimensional array
Keras	Open source library to work with neural networks
Os	To travel through local system and fetch data in specified location
Matplot	Open source library for implementing graph
Conv2d	Used to move the layer over the input image
DenseNet	The node of the layer to connected with all the node in the previous layer

iterrows () function. Import os header file to fetch the location inside the local system (Table 1).

5.1 Proposed Methodology

It is a deep education algorithm which is used while dealing with images. It can be used specifically for recognition images pixels and processing of images. It is a software pattern which has three layers

1. Inner layer
2. Hidden layers
3. Output layer

The nodes in CNN are called neurons which has weight and biases. All the neurons will get various inputs and fetch a highest sum of all of them, which is given in an

activation function, and result is got out as a response; convolutional layer is the primary layer in the exemplary and is used to fetch out of the different featured from the input. In convolutional layer, the mathematical action is done in between the image and the specified size $n \times n$. The edge and corners are the output from this layer and is fed as a input to the next layers.

In common, the next layer to convolutional coating is the pooling layer. The main goal of this level is to reduce the feature map size and hence reducing the processing cost. This is performed by diminishing the associations among layers and autonomously works on each component map. Contingent on strategy utilized, there are a few sorts of pooling tasks.

In max pooling, the biggest component is taken from highlight map. Normal pooling ascertains the normal of the components in a predefined measured image segment. The all out amount of the components in the predefined area is registered in sum pooling. The pooling layer generally fills in as a scaffold among the fully connected layer and convolutional layer.

Max pooling: The most extreme pixel estimation of the bunch is chosen.

Min pooling: The base pixel estimation of the group is chosen.

Normal pooling: The normal estimation of the multitude of pixels in the clump is chosen.

Built a model with four layers and one dense output layer and train the model with the train data. The number of epoch and steps per epoch can be set depending on the amount of data to be trained. The class model for this project is binary as there are only two outputs possible. The summary can also be seen with the help of summary function.

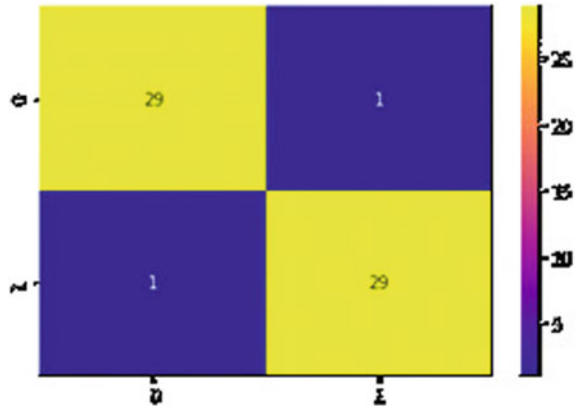
While training the model, specify the image processing parameter like the size to which all the inputs have to be reduced. This is an important step to make the model work more efficient than the existing models. For effective processing, use ImageData Generator while training from the `scat,ch`. Also check if the number of classes in the dataset is only two.

The model can be validated with the help of test dataset. Validating the model will reveal us how good the model fits the dataset. A good practice is try to reduce the loss instead of focusing in improving the accuracy. The 20% of the dataset can be used for validating the model. It is very important to verify if the model over fit the dataset. In such case, the loss will be very high.

Over fitting: Performance is good on the working out dataset, do poor on other data.

Under fitting: Performance poor on the exercise dataset and do poor on other data.

All these can be changed by tweaking the values of the parameters while testing.

Fig. 4 Confusion matrix

6 Analysis of the Proposed Scheme

The knowledge about the project can be gained from the existing models. The COVID-19 sufferers are detected with the benefit of patients X-ray image. The main drawback is that these model takes the image as a whole and processes them to predict the result. This can be made even more efficient by reducing all the image sizes evenly leaving the unwanted extra details in the image.

Their existing system helps in knowing some of the basic and very important factors that are essential to begin the model. In this existing model, the upper-body X-ray image of in good physical person, COVID patients, bacterial pneumonia, and virus pneumonia person as are given as a contribution to the pre-trained neural network and the output is predicted. The results can be seen in confusion matrix in Fig. 4.

To compare the various computing deep learning algorithm and to find the results in accuracy are shown in Fig. 5, and to find the best algorithm out of all of them in CNN.

7 Performance Analysis of the Proposed Scheme

Calculating the loss with the help of epochs and value loss in a graph will represent how well the model fits to the data. There spread of COVID all over the world and all the people are affected by the deadly virus. Though the vaccines are available, the spread of virus is no way reduced. The main reason for more spread is that people are not aware whether they are affected by COVID or not. Though the COVID tests can help to identify the patients, it is taking a long time for the results and it is enough time for the virus to spread from one to other. Though lockdown, wearing mask and getting vaccinate is being followed, the spread of virus exists (Fig. 6).

Fig. 5 Comparing the accuracy of deep learning algorithm

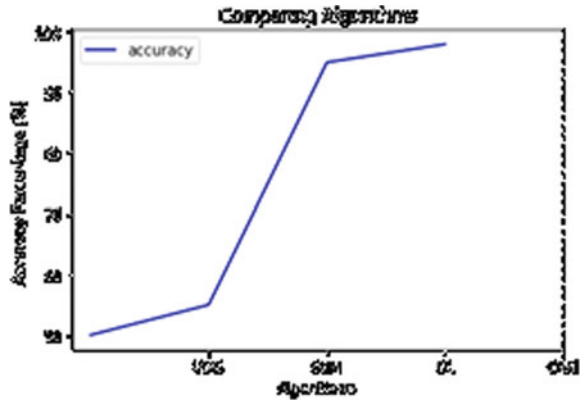
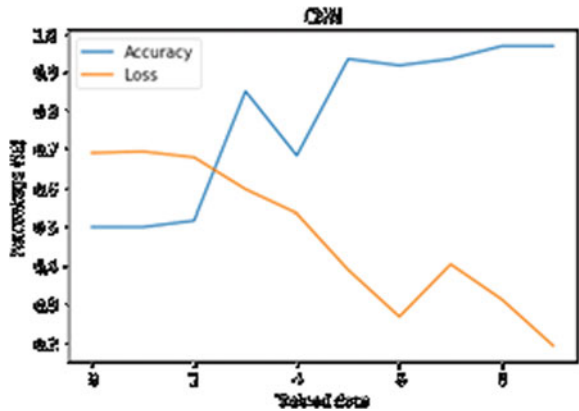


Fig. 6 CNN loss and accuracy



The accuracy of the validated CNN model can also be made as a graph with the help of epochs and value accuracy which is represented in Fig. 6. The point that can be speeded up is the duration of identifying COVID patients with the COVID test can be reduced. Average of 14 people gets in contact with the patient for every one hour. Reducing the duration of finding result by speed up alternate test can be finding to reduce the spread. This can be made possible with the benefit of deep learning. Convolutional network is a deep learning algorithm which is commonly used to work with images.

This can help in detecting and identifying the COVID-affected patients very quickly. All need the upper-body X-ray image of the person. If the image of the patient upper-body X-ray is feed to the ideal, it can predict the result in no time, and thus, the patients are identified quickly, isolate, and treat the patient very quickly.

8 Conclusion

Deep learning may be a versatile series of neural network learning techniques. Neural networks inspired a programming model that enables a machine to be instructed based on empirical evidence. Convolutional networks are a form of profound neural network that is widely used to analyze visual images in deep learning. Biological processes influenced convolutional networks, in which the property pattern between neurons creates the animal cortical area's consortium. Biological processes inspired convolutional networks, in which the property configuration between neurons looks a lot like the arrangement of the innate cortical region. Any input image can move through a series of convolution layers with filters, pooling, totally connected layers (FC), and associated softmax perform to classify an object with deep learning CNN models. One among the most elements of neural networks is convolutional neural networks. They are made up of neurons with weights and prejudices that can be learned. Every somatic cell receives a variety of inputs and computes a weighted average of them before passing it through associate activation and responding with an output. A convolutional neural network may be a neural network that has one additional convolutional layer and is employed in the main for image process, classification, segmentation, and additionally for alternative autocorrelated information. Then, CNN works well on a picture because it takes all the pixels and generates a pattern from the testing information. The pattern generated is applied to the coaching information to come up with the high accuracy result.

References

1. Ozturk T, Talo M, Yildirim EA, Baloglu UB, Yildirim O, Acharya UR (2020) Automated detection of COVID-19 cases using deep neural networks with X-ray images. *Comput Biol Med* 103792. PMID: 32568675
2. Lee EY, Ng MY, Khong PL (2020) COVID-19 pneumonia: what has CT taught us? *Lancet Infect Dis* 20(3):384–385
3. Pandiaraja P, Sharmila S (2020) Optimal routing path for heterogeneous vehicular adhoc network. *Int J Adv Sci Technol* 29(7s):1762–1771
4. He K, Zhang X, Ren S, Sun J (2016) Identity mappings in deep residual networks. In: *European conference on computer vision*, pp 630–645
5. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*
6. Zheng C, Deng X, Fu Q et al (2020) Deep learning-based detection for COVID-19 from chest CT using weak label. *medRxiv*
7. Majeed T, Rashid R, Ali D, Asaad A (2020) Covid-19 detection using CNN transfer learning from X-ray images. *medRxiv*
8. Xu X, Jiang X, Ma C, Du P, Li X, Lv S, Yu L, Chen Y, Su J, Lang G, Li Y, Zhao H, Xu K, Ruan L, Wu W (2020) Deep learning system to screen coronavirus disease 2019 pneumonia. *arXiv:2002.09334*
9. Deepika S, Pandiaraja P (2013) Ensuring CIA triad for user data using collaborative filtering mechanism. In: *2013 international conference on information communication and embedded systems (ICICES)*, pp 925-928. <https://doi.org/10.1109/ICICES.2013.6508262>

10. El Asnaoui K, Chawki Y (2020) Using X-ray images and deep learning for automated detection of coronavirus disease. *J Biomol Struct Dyn* 1–12
11. Deepa N, Pandiaraja P (2020) Hybrid context aware recommendation system for E-health care by merkle hash tree from cloud using evolutionary algorithm. *Soft Comput* 24:7149–7161
12. Chollet F (2020) Xception: deep learning with depth wise separable convolutions. In: 2017 IEEE CVPR, pp 1251–1258
13. Minaee S, Kafieh R, Sonka M, Yazdani S, Soufi GJ (2020) Deep-covid: predicting covid-19 from chest X-ray images using deep transfer learning. arXiv preprint [arXiv:2004.09363](https://arxiv.org/abs/2004.09363)
14. Koo HJ, Lim S, Choe J, Choi SH, Sung H, Do KH (2018) Radiographic and CT features of viral pneumonia. *Radiographics* 38(3):719–739
15. Pandiaraja P, SanthanaHari S, Suriya S, Karthikeyan S (2020) Convolutional neural network for solid waste segregation and management. *Int J Adv Sci Technol* 29(7s):1661–1668
16. Huang C, Wang Y, Li X, Ren L, Zhao J, Hu Y, Zhang L, Fan G, Xu J, Gu X, Cheng Z (2020) Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. *Lancet* 395(10223):497–506
17. Holshue ML, DeBolt C (2020) First case of 2019 novel coronavirus in the United States. *N Engl J Med* 328:929–936
18. Narin A, Kaya C, Pamuk Z (2020) Automatic detection of coronavirus disease (covid-19) using X-ray images and deep convolutional neural networks. arXiv preprint [arXiv:2003.10849](https://arxiv.org/abs/2003.10849)
19. Pandiaraja P, Aravinthan K, Lakshmi Narayanan R, Kaaviya KS, Madumithra K (2020) Efficient cloud storage using data partition and time based access control with secure AES encryption technique. *Int J Adv Sci Technol* 29(7s):1698–1706
20. Sumathi K, Pandiaraja P (2020) Dynamic alternate buffer switching and congestion control in wireless multimedia sensor networks. *Peer-to-Peer Netw Appl* 13:2001–2010
21. Zhang Y et al (2020) COVID-DA: deep domain adaptation from typical pneumonia to COVID-19. [arXiv:2005.01577](https://arxiv.org/abs/2005.01577)
22. Pandiaraja P, Manikandan J (2015) Web proxy based detection and protection mechanisms against client based HTTP attacks. In: 2015 international conference on circuits, power and computing technologies [ICCPCT-2015], pp 1–6. <https://doi.org/10.1109/ICCPCT.2015.7159344>
23. Yoon SH, Lee KH (2020) Chest radiographic and CT findings of the 2019 novel coronavirus disease (COVID-19): analysis of nine patients treated in Korea. *Korean J Radiol*
24. Deng J et al (2009) Imagenet: a large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition, pp 248–255
25. Huang G, Liu Z, Weinberger KQ (2016) Densely connected convolutional networks. arXiv preprint [arXiv:1608.06993](https://arxiv.org/abs/1608.06993)
26. Deepa N, Pandiaraja P (2021) E health care data privacy preserving efficient file retrieval from the cloud service provider using attribute based file encryption. *J Ambient Intell Human Comput* 12:4877–488
27. Khobahi S, Agarwal C, Soltanalian M (2020) Coronet: a deep network architecture for semi-supervised task-based identification of covid-19 from chest X-ray images. medRxiv. <https://doi.org/10.1101/2020.04.14.20065722>
28. Shen D, Wu G, Suk HI (2017) Deep learning in medical image analysis. *Annu Rev Biomed Eng* 19:221–248
29. Rajpurkar P, Irvin J (2017) Chexnet: radiologist-level pneumonia detection on chest X-rays with deep learning. arXiv preprint [arXiv:1711.05225](https://arxiv.org/abs/1711.05225)
30. Luz E et al (2020) Towards an effective and efficient deep learning model for covid-19 patterns detection in X-ray images. [arXiv:2004.05717](https://arxiv.org/abs/2004.05717)
31. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521:436–444
32. Rasheed J, Jamil A, Hameed AA, Aftab U, Aftab J, Shah SA et al (2020) A survey on artificial intelligence approaches in supporting frontline workers and decision makers for COVID-19 pandemic. *Chaos Solitons Fractals* 110337. PMID: 33071481
33. Pandiaraja P, Deepa N (2019) A novel data privacy-preserving protocol for multi-data users by using genetic algorithm. *Soft Comput* 23:8539–8553

34. Zu ZY, Jiang MD, Xu PP, Chen W, Ni QQ, Lu GM, Zhang LJ (2020) Coronavirus disease 2019 (COVID-19): a perspective from China. *Radiology*. <https://doi.org/10.1148/radiol.2020200490>
35. Kong W, Agarwal PP (2020) Chest imaging appearance of COVID-19 infection. *Radiol Cardiothorac Imaging*
36. Li Y, Xia L (2020) Coronavirus disease 2019 (COVID-19): role of chest CT in diagnosis and management. *Am J Roentgenol*
37. Pandiaraja P, Parasuraman S (2015) Applying secure authentication scheme to protect DNS from rebinding attack using proxy. In: 2015 international conference on circuits, power and computing technologies [ICCPCT-2015], pp 1–6. <https://doi.org/10.1109/ICCPCT.2015.7159255>
38. Zheng C, Deng X, Fu Q, Zhou Q, Feng J, Ma H, Wang X (2020) Deep learning-based detection for COVID-19 from chest CT using weak label. medRxiv. <https://doi.org/10.1101/2020.03.12.20027185>
39. Barstugan M, Ozkaya U, Ozturk S (2020) Coronavirus (COVID-19) classification using CT images by machine learning methods. arXiv preprint [arXiv:2003.09424](https://arxiv.org/abs/2003.09424)
40. Gaal G, Maga B, Lukács A (2020) Attention U-Net based adversarial architectures for chest X-ray lung segmentation. arXiv preprint [arXiv:2003.10304](https://arxiv.org/abs/2003.10304)
41. Pan SJ, Yang Q (2009) A survey on transfer learning. *IEEE Trans Knowl Data Eng* 22(10):1345–1359
42. Wang S, Kang B, Ma J et al (2020) A deep learning algorithm using CT images to screen for corona virus disease (COVID-19). medRxiv
43. Chouhan V, Singh SK, Khamparia A et al (2020) A novel transfer learning based approach for pneumonia detection in chest X-ray images. *Appl Sci* 10(2):559
44. Faust O, Hagiwara Y, Hong TJ, Lih OS, Acharya UR (2018) Deep learning for healthcare applications based on physiological signals: a review. *Comput Methods Programs Biomed* 161:1–13. PMID: 29852952
45. Osman AH, Aljahdali HMA (2020) An effective of ensemble boosting learning method for breast cancer virtual screening using neural network model. *IEEE Access* 8:39165–39174
46. Sedik A, Hammad M, Abd El-Samie FE, Gupta BB, Abd El-Latif AA (2021) Efficient deep learning approach for augmented detection of coronavirus disease. *Neural Comput Appl* 1–18. PMID: 33487885
47. Kamal K, Yin Z, Wu M, Wu Z (2021) Evaluation of deep learning-based approaches for COVID-19 classification based on chest X-ray images. *Signal Image Video Process* 1–8
48. Singh RK, Pandey R, Babu RN (2021) COVIDScreen: explainable deep learning framework for differential diagnosis of COVID-19 using chest X-rays. *Neural Comput Appl* 1–22. PMID: 33437132
49. Talo M, Yildirim O, Baloglu UB, Aydin G, Acharya UR (2019) Convolutional neural networks for multi-class brain disease detection using MRI images. *Comput Med Imaging Graph* 78:101673. PMID: 31635910

Classification of Medical Health Records Using Convolutional Neural Networks for Optimal Diagnosis



M. H. Chaithra and S. Vagdevi

Abstract Pneumonia is considered to be one of the lungs affecting inflammation for small air sacs. Dry cough, chest pain, fever, and breathing difficulty are some of the common symptoms during this situation. The seriousness of the condition of the patient is variable based on several parameters. Viruses, bacteria, and by other microorganisms usually cause pneumonia. Some of the risk factors during this situation are: cystic fibrosis, chronic obstructive pulmonary disease (COPD), asthma, diabetes, and heart failure. Sometimes a weak immune system may also increase the severity of the situation. The medical diagnostics using machine learning powered by computer vision and deep learning will help us to extract useful information by filtering out the non-essential and insignificant information from the diagnosis report. Computer vision, neural networks, and artificial intelligence methods like convolutional neural network will lead to identify and extract the useful information from the diagnosis report, and in turn, it will help to assist in medical diagnosis. In this regard, the main objective of this work is to classify disease based on symptoms. Clinical and laboratory symptoms are considered as the basic for this investigation.

Keywords Machine learning · Computer vision · Artificial neural networks · Convolutional neural networks · Medical diagnostics · Data augmentation · Image acquisition

1 Introduction

As per the research released by the Indian Journal of Medical Research under the title “Doctor Population ratio of India—The Reality,” it has been estimated that six

M. H. Chaithra (✉)

Department of Computer Science and Engineering, REVA University, Bangalore, India

Visvesvaraya Technological University, Belagavi, Karnataka, India

e-mail: chaithra.mh14@gmail.com

S. Vagdevi

Department of Computer Science and Engineering, City Engineering College, Bangalore, India

lakh doctors and twenty lakh nurses shortage of medical personnel in India. India is planning to establish two hundred new medical colleges in next decade to meet the above requirement. Also, the cost of medical treatment is increasing for a common man in India and 65% of the health expenditure is borne by the individual itself, and as per recent release of the data by the government of India, it has been estimated that the medical expenses push 57 million people into poverty each year.

The medical diagnostics using machine learning powered by computer vision and deep learning will help us to extract useful information by filtering out the non-essential and insignificant information from the diagnosis report [1]. Computer vision, neural networks, and artificial intelligence methods like convolutional neural network will lead to identify and extract the useful information from the diagnosis report, and in turns, it will help to assist in medical diagnosis. We will train and develop the medical diagnostics tool which will help organization or government or user which will assist doctors/medical personnel in medical diagnosis.

2 Background

The National Electronic Health Records Survey (NEHRS) is an annual comprehensive survey of employed, office-based physicians. Usually based on the role of diet and prescribed exercise, the health risks are evaluated and research will be conducted. Recently, many researchers have achieved promising results based on electronic database and applying computational techniques. At the same time, secured data and maintaining patient's privacy are also primary concerns while maintaining health records electronically. This research article provides an optimal method to identify a specific disease by suitable computational methods and also justifies the reliability on developed system.

3 Objectives

As per the above scope, the following objectives are defined in this research work:

- Medical diagnostics using machine learning
- Developing a medical diagnostics tool
- Medical diagnostics tool powered by machine learning and deep learning will high prediction capability than the traditional models
- Identifying and extract the most critical/important information from the diagnostic
- Reducing the manual touchpoint while performing the model diagnostic.

4 Proposed Process Flow

To carry out the proposed research work, the resources needed are—artificial neural network, computer vision, and Python. Hardware requirements are 32 GB RAM, 1 TB Hard Disk, Window/Linux Machine. The potential challenges and risks involved are different sources of data which will have different patterns and quality of data which usually lot of efforts to prepare and clean for analysis [2]. Privacy of the data is also one of the challenges in such types of domain. Figure 1 presents the overall flow of the proposed model.

Around 6000 JPEG X-ray images are considered for detecting pneumonia condition. Training, testing, and validation are the stages in which different subfolders of images are distributed [3]. Anterior–posterior chest images are selected from pediatric patients for this study. Clinical and laboratory symptoms are considered which selecting chest images for the investigation [4]. Several chest radiographs are filtered to remove images with noise, poor quality, or unreadable data. Finally, the filtered images are certified by experts before being used for training purpose for our model. In this phase, the grading errors are recorded and discarded from the training database.

5 Methodology

5.1 Dataset Collection

One of the major potential challenges for this work is to obtain relevant medical data. As mentioned earlier, around 6000 JPEG images are been considered from unique patients for this study. Based on the associated radiology reports, the text contents are extracted and used for classification using preprocessing phases of language processing tools [5]. A unique labeling process is adopted to disambiguate and group the images according to the clinical text data as per the proposal in the arti-

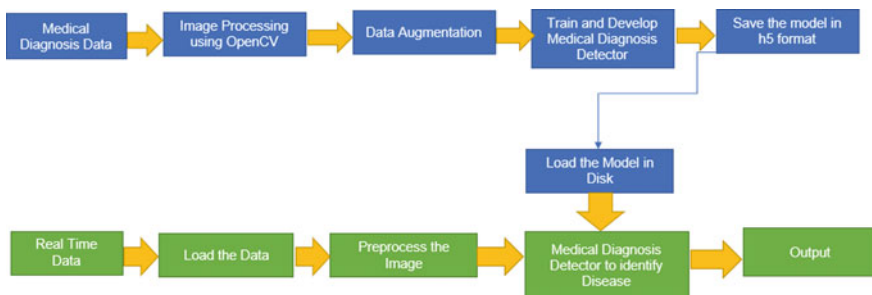


Fig. 1 Proposed flow diagram indicating all involved modules

cle, “ChestX-ray8: Hospital-scale Chest X-ray database and benchmarks on weakly supervised classification and localization of common thorax diseases” (Wang et al.). This dataset includes 12 zip files, and each of them is of size 2–4 GB. The technologies used are: Keras, Python, Spyder, Jupyter, OpenCv, TensorFlow, and image acquisition through CNN datasets.

The typical text preprocessing steps involved are: removal of white space, expanding the contraction, removing noise, special character, normalizing all text to lower case, finding the maximum length of the text, tokenization, stop word removal, and stemming/lemmatization.

5.2 *Preprocessing*

Standardization of the features is conducted by standardizing pixel values among the whole database. This action is applied for each column in a tabular database [6]. Feature-centric and feature standard normalization parameters are used to standardize the entire image data generation class. This process is monitored closely in order to avoid multiple arguments with same effect. Otherwise, the redundant entries need to be filtered which will be an added effort. Typical image processing algorithms are applied on these digitized images. There are many applications of digital image processing algorithms as compared to analog processing. Majority of digital image processing algorithms help in enhancing image features by eliminating noise or skewed images. These enhanced image parameters considerably improve in developing artificial intelligent computer models. Typical image processing phases include—reading image, resizing it, de-noise (if any), normalize it, segment, and smooth edges as per the needs.

6 **Model Building**

The typical neural network model and convolutional neural network model are shown in Figs. 2 and 3. Convolutions are meant to extract key features from the input images. By learning image features, they ensure the relationship among pixels of input images [7]. The two inputs such as image matrix and kernel or filter are considered for a mathematical operation. We all know how to generate the volume dimension as output from an image matrix of dimension using relevant filter [8]. The convolution of image matrix multiplies with filter matrix to generate feature map. Strides in convolution layer are significant.

This section provides a complete overview on the developed model along with the code samples. Figures 4 and 5 show the code for preprocessing steps and visualization steps.

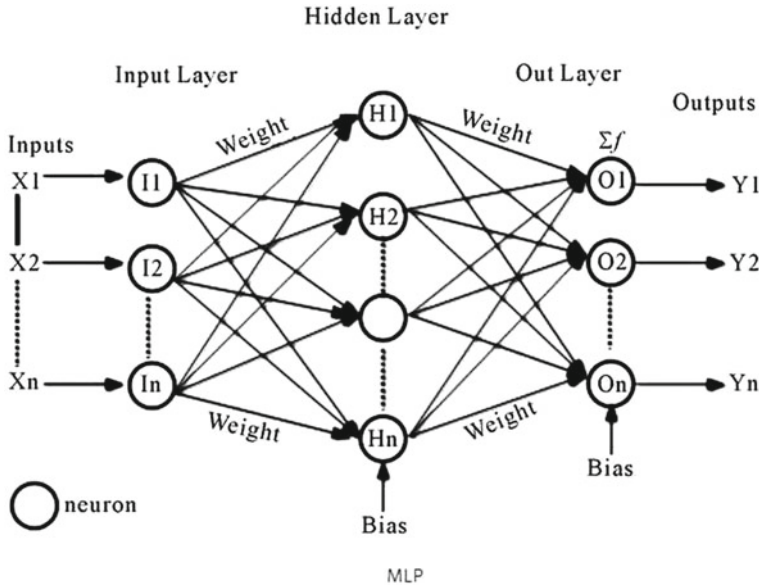


Fig. 2 Simple neural network architecture

It is noticed that the feature map size is smaller than the input size. The feature map has to be avoided from shrinking [9] with the help of padding process. Zero-valued pixels are added around the input in order to avoid shrinking of feature map. This will ensure that the spatial size remains constant. Padding confirms improvement in the performance, and kernel size is constantly maintained. When the input images are large, then the number of arguments will be reduced by pooling. By retaining the key features, dimension reduction happens through downsampling. Various downsampling methods are maxpooling, average pooling, and sum pooling [10]. Suitable bias values are applied to ensure an efficient activation function. When the derivatives are steeper, several neurons will get destroyed which results in a passive network. When the epoch is consuming huge time to run, then it is decomposed into batches. Binary cross-entropy measures are used to average the class-wise errors. Adam optimizer is used to update network weights iterative based on training data. It pursues a single learning rate for all weights updates. The learning rate is undisturbed during training process. Convolution layer is basically a feature detector that automatically tries to learn to filter out the not needed information. Pooling layers reduce the memory size required for processing and also detect object characteristics at some unusual places.

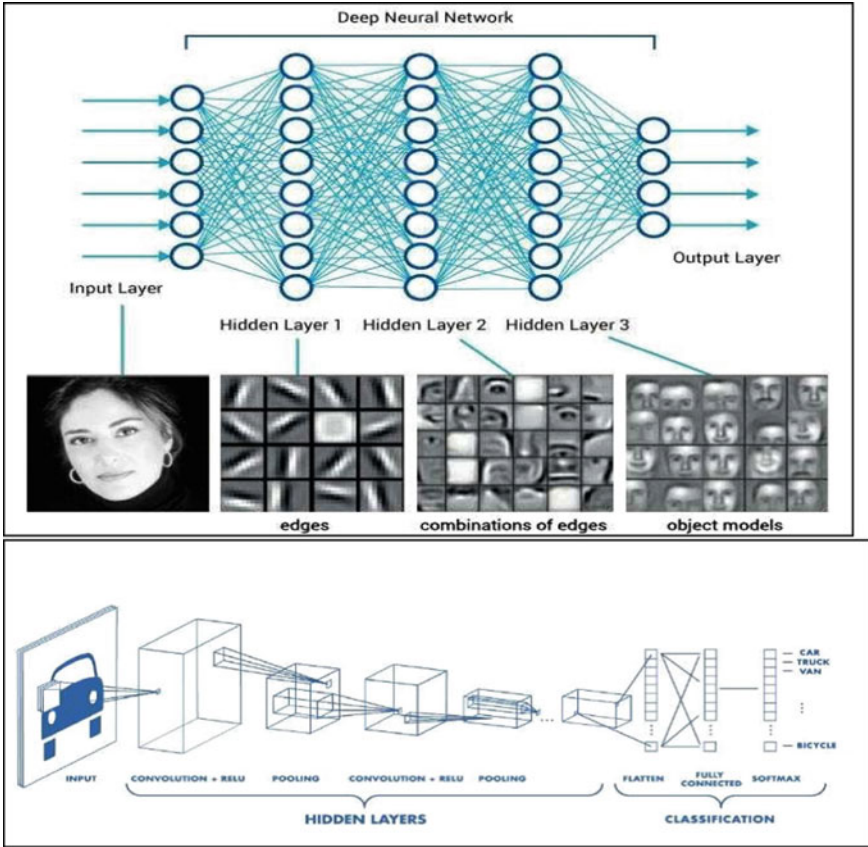


Fig. 3 Convolutional neural network architecture

```
import os
import numpy as np # Linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)

import matplotlib.pyplot as plt
import seaborn as sns
import keras
from keras.models import Sequential
from keras.layers import Dense, Conv2D, MaxPool2D, Flatten, Dropout, BatchNormalization
from keras.preprocessing.image import ImageDataGenerator
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report, confusion_matrix
from keras.callbacks import ReduceLROnPlateau
import cv2
```

Fig. 4 A sample code showing preprocessing and visualization

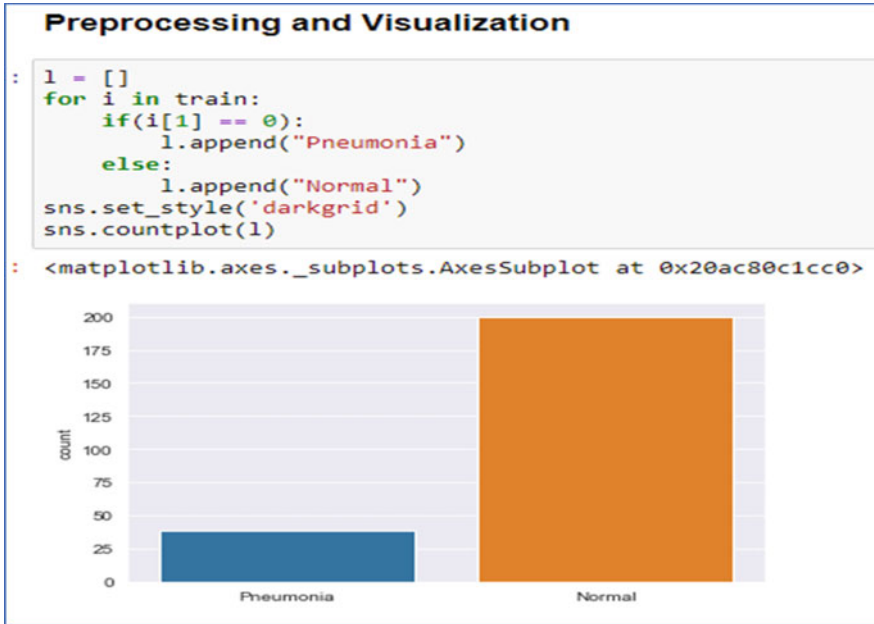


Fig. 5 Graph showing the status of pneumonia

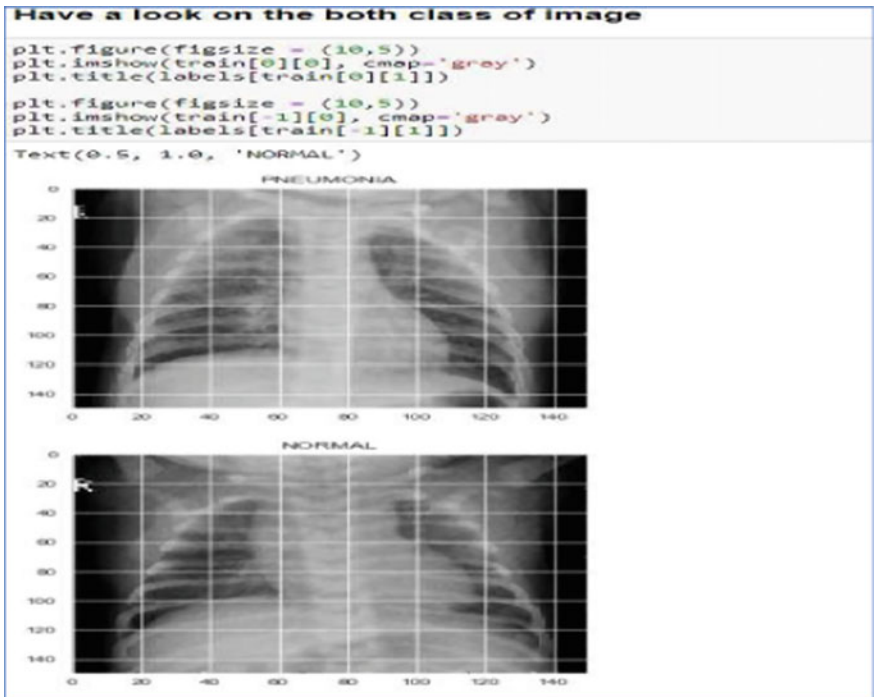


Fig. 6 Sample images of affected lungs

```

Normalize the data

x_train = np.array(x_train) / 255
x_val = np.array(x_val) / 255
x_test = np.array(x_test) / 255

Resizing the data to bring them into proper format

x_train = x_train.reshape(-1, img_size, img_size, 1)
y_train = np.array(y_train)

x_val = x_val.reshape(-1, img_size, img_size, 1)
y_val = np.array(y_val)

x_test = x_test.reshape(-1, img_size, img_size, 1)
y_test = np.array(y_test)

Augmenting the data

datagen = ImageDataGenerator(
    featurewise_center=False, # set input mean to 0 over the dataset
    samplewise_center=False, # set each sample mean to 0
    featurewise_std_normalization=False, # divide inputs by std of the dataset
    samplewise_std_normalization=False, # divide each input by its std
    zca_whitening=False, # apply ZCA whitening
    rotation_range = 30, # randomly rotate images in the range (degrees, 0 to 180)
    zoom_range = 0.2, # Randomly zoom image
    width_shift_range=0.1, # randomly shift images horizontally (fraction of total width)
    height_shift_range=0.1, # randomly shift images vertically (fraction of total height)
    horizontal_flip = True, # randomly flip images
    vertical_flip=False) # randomly flip images

datagen.fit(x_train)

For the data augmentation,Following parameters havebeen seilected

    Randomly rotate some training images by 30 degrees
    Randomly Zoom by 20% some training images
    Randomly shift images horizontally by 10% of the width
    Randomly shift images vertically by 10% of the height
    Randomly flip images horizontally. Once our model is ready, we fit the training dataset.

```

Fig. 7 Sample code showing data normalization, resizing, and augmentation

```

Architecture for training the model

model = Sequential()
model.add(Conv2D(32, (3,3), strides = 1, padding = 'same', activation = 'relu', input_shape = (150,150,1)))
model.add(BatchNormalization())
model.add(MaxPool2D((2,2), strides = 2, padding = 'same'))
model.add(Conv2D(64, (3,3), strides = 1, padding = 'same', activation = 'relu'))
model.add(Dropout(0.1))
model.add(BatchNormalization())
model.add(MaxPool2D(2,2), strides = 2, padding = 'same'))
model.add(Conv2D(64, (3,3), strides = 1, padding = 'same', activation = 'relu'))
model.add(BatchNormalization())
model.add(MaxPool2D(2,2), strides = 2, padding = 'same'))
model.add(Conv2D(128, (3,3), strides = 1, padding = 'same', activation = 'relu'))
model.add(Dropout(0.2))
model.add(BatchNormalization())
model.add(MaxPool2D(2,2), strides = 2, padding = 'same'))
model.add(Conv2D(256, (3,3), strides = 1, padding = 'same', activation = 'relu'))
model.add(Dropout(0.2))
model.add(BatchNormalization())
model.add(MaxPool2D(2,2), strides = 2, padding = 'same'))
model.add(Flatten())
model.add(Dense(units = 128, activation = 'relu'))
model.add(Dropout(0.2))
model.add(Dense(units = 1, activation = 'sigmoid'))
model.compile(optimizer = "rmsprop", loss = 'binary_crossentropy', metrics = ['accuracy'])
model.summary()

```

Fig. 8 Code sample to show the process of training the proposed model

7 Code Snippet

All the required class definitions and visualization steps essential for the model are shown. Figure 6 shows images of both normal and pneumonia-affected lungs images. The deviations obtained in these affected images are evident of the percentage of infection through increased number of epochs of CNN. The code for data normalization which is responsible for noise elimination and filtering is shown in Fig. 7. Thus, figure also shows resizing and augmentation code. Figure 8 illustrates the steps followed for training the model. We classified trainable and non-trainable parameters from the input file and identify the percentage of data samples collected. Figure 9 lists the parameters for training the model.

8 Analysis of Model Performance

The proposed work presents the optimal method of analyzing patient's health records in the form of images. Through CNN, the training accuracy and accuracy of the validated results are checked. Figure 10 shows the graph with promising results, and it is evident that the method followed is reliable. Another graph is also shown with very minimal loss rate from the considered datasets.

9 Conclusion and Future Scope

The proposed solution project would be used by the organization/government authorities/medical authorities to reduce the workload of overloaded medical personnel and provide medical facilities to everyone at affordable cost.

Pneumonia is considered to be one of the serious statuses of health which leads to considerable proportion of mortality. This status can be controlled by early diagnosis with some computational techniques. Among various diagnostic procedures, chest X-rays are considered to be a reliable tool for screening and examination. Even though considerable imaging equipments are available, shortage of experts to infer the images is an added challenge. This work facilitates in proposing additional procedure for early detection of the disease through clinical and laboratory evidences of chest X-ray images.

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 150, 150, 32)	320
batch_normalization_1 (Batch Normalization)	(None, 150, 150, 32)	128
max_pooling2d_1 (MaxPooling2D)	(None, 75, 75, 32)	0
conv2d_2 (Conv2D)	(None, 75, 75, 64)	18496
dropout_1 (Dropout)	(None, 75, 75, 64)	0
batch_normalization_2 (Batch Normalization)	(None, 75, 75, 64)	256
max_pooling2d_2 (MaxPooling2D)	(None, 38, 38, 64)	0
conv2d_3 (Conv2D)	(None, 38, 38, 64)	36928
batch_normalization_3 (Batch Normalization)	(None, 38, 38, 64)	256
max_pooling2d_3 (MaxPooling2D)	(None, 19, 19, 64)	0
conv2d_4 (Conv2D)	(None, 19, 19, 128)	73856
dropout_2 (Dropout)	(None, 19, 19, 128)	0
batch_normalization_4 (Batch Normalization)	(None, 19, 19, 128)	512
max_pooling2d_4 (MaxPooling2D)	(None, 10, 10, 128)	0
conv2d_5 (Conv2D)	(None, 10, 10, 256)	295168
dropout_3 (Dropout)	(None, 10, 10, 256)	0
batch_normalization_5 (Batch Normalization)	(None, 10, 10, 256)	1024
max_pooling2d_5 (MaxPooling2D)	(None, 5, 5, 256)	0
flatten_1 (Flatten)	(None, 6400)	0
dense_1 (Dense)	(None, 128)	819328
dropout_4 (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 1)	129
=====		
Total params: 1,246,401		
Trainable params: 1,245,313		
Non-trainable params: 1,088		

Fig. 9 Status of total parameters considered for training the model

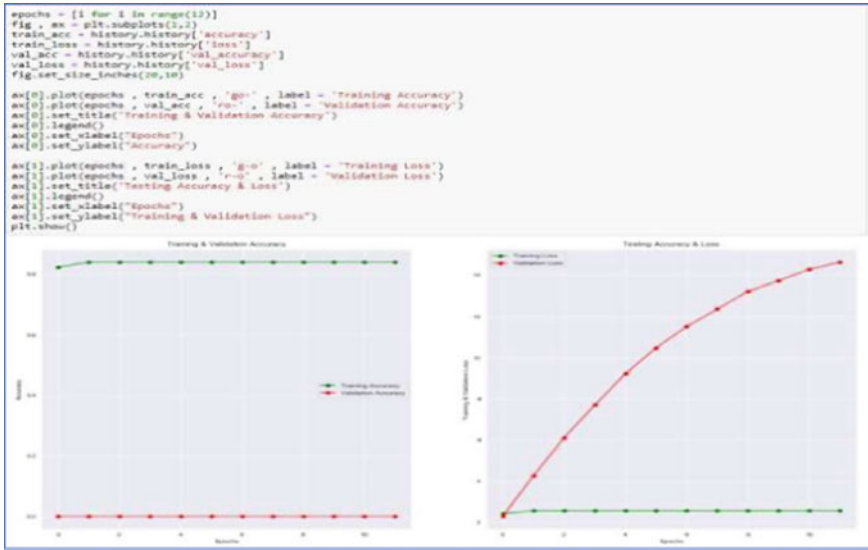


Fig. 10 Performance of the model developed

References

1. Aydogdu M, Ozyilmaz E, Aksoy H, Gursel G, Ekim N (2010) Mortality prediction in community-acquired pneumonia requiring mechanical ventilation; values of pneumonia and intensive care unit severity scores. *Tuberk Toraks* 58(1):25–34
2. Berbaum K, Franken EA Jr, Smith WL (1985) The effect of comparison films upon resident interpretation of pediatric chest radiographs. *Investig Radiol* 20(2):124–128
3. Cherian T, Mulholland EK, Carlin JB, Ostensen H, Amin R, de Campo M, Greenberg D, Lagos R, Lucero M, Madhi SA et al (2005) Standardized interpretation of pediatric chest radiographs for the diagnosis of pneumonia in epidemiological studies. *Bull World Health Organ* 83(5):353–359
4. Kingma D, Ba JA (2014) A method for stochastic optimization. arXiv preprint [arXiv:1412.6980](https://arxiv.org/abs/1412.6980)
5. <https://data.mendeley.com/datasets/rscbjbr9sj/2>
6. Collen MF, Slack WV, Bleich HL (2015) Medical databases and patient record systems. In: Collen M, Ball M (eds) *The history of medical informatics in the United States*. Health informatics. Springer, London. https://doi.org/10.1007/978-1-4471-6732-7_4
7. Vimalachandran P, Wang H, Zhang Y (2015) Securing electronic medical record and electronic health record systems through an improved access control. In: Yin X, Ho K, Zeng D, Aickelin U, Zhou R, Wang H (eds) *Health information science. HIS 2015. Lecture notes in computer science*, vol 9085. Springer, Cham. https://doi.org/10.1007/978-3-319-19156-0_3
8. Cahill JE, Gilbert MR, Armstrong TS (2014) Personal health records as portal to the electronic medical record. *J Neurooncol* 117:1–6. <https://doi.org/10.1007/s11060-013-1333-x>
9. Kruse CS, Stein A, Thomas H et al (2018) The use of electronic health records to support population health: a systematic review of the literature. *J Med Syst* 42:214. <https://doi.org/10.1007/s10916-018-1075-6>
10. Smaradottir BF, Fensli RW (2020) User experiences and satisfaction with an electronic health record system. In: Ahram T, Falcão C (eds) *Advances in usability and user experience. AHFE 2019. Advances in intelligent systems and computing*, vol 972. Springer, Cham. https://doi.org/10.1007/978-3-030-19135-1_8

Smart Farming Using IoT Sensors



J. Y. Srikrishna and J. Sangeetha

Abstract Agriculture is the backbone of India. Agriculture is the sector where water usage is more with irrigation accounting 75% of global water usage. If we don't make any improvement in efficiency of usage of water, it is expected that usage of water for agriculture will increase by 20%. As population is increasing day-by-day, agriculture is becoming more important factor as it feeds many of the lives and we should be blessed for that. In the traditional approach, farmer will not get to know how much amount of water does the plant needs, so plants will not get required amount of water because of which we may end up in either more or less water supplied to the plants. In this research work, by using smart farming technology using Internet of Things (IoT) we will get to know soil moisture level, according to the soil moisture level the farmer can supply required amount of water through water pump. By this work we can overcome two extremes of the problem either reducing the wastage or too much consumption of water. Once water resource is used efficiently, the next stage is to check whether the banana plant is healthy or not. The banana plant health checkup is monitored through its leaf. Here, we are checking whether the leaf is diseased or not using SVM algorithm. From this research work, we are saving the water resource and plant health state in its initial stage. Thus, we have improved the efficiency of usage of water to the banana plant.

Keywords Internet of Things · Arduino Uno · Sensors · Node MCU · SVM algorithm

J. Y. Srikrishna (✉) · J. Sangeetha
Department of Computer Science and Engineering, M S Ramaiah Institute of Technology,
Bangalore, India
e-mail: rikrishnajagglar@gmail.com

J. Sangeetha
e-mail: sangeethakirank@msrit.edu

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
D. P. Agrawal et al. (eds.), *Cyber Security, Privacy and Networking*, Lecture Notes
in Networks and Systems 370, https://doi.org/10.1007/978-981-16-8664-1_23

259

1 Introduction

In the development of the agricultural nation as India, agri-industry expects vital employment. The country's changes have been continually undermined by agricultural problems. Smart agriculture, which involves modernizing current conventional agricultural systems, is the leading solution to this issue. To assist with field work, most farmers use tractors and other motorized machinery. Tractors, like plows, are much bigger, allowing farmers to grow more food in less time. Farmers have always attempted to make the most of their resources, but modern farmers have been able to engage in sustainable farming methods such as conservation, restoration, and moderation because to continually increasing technology.

The Smart Agriculture system helps to make agriculture more involved in the field of IoT and robotics [1]. A scheduled IoT water system based on Arduino Uno is proposed for modernization and increased development profitability. The fundamental objective of this work is to improve progress in obtaining the correct amount of water in the soil at the appropriate period, for which most producers spend their money on the fields. A useful water organization should be generated to decrease the complex nature of the water structuring circuit.

This design suggested that by sending the data to the sensor and measuring the required amount of water, it is produced. The device proposed consists of a base station and sensors. Three sensors, including the humidity and dirt moisture, temperature and period of sunlight, continuously collect the information from the base station. The proposed structure helps to assess the amount of water required for the water system. The real versatility within the system is precision agriculture (PA) [2] with distributed data processing, which increases the use of water fertilizers while raising plant yields and also helps to break down climatic conditions in the field.

The importance of agricultural irrigation in plant production was discussed by Sushanth et al. [1]. It is one of the main variables for the survival of human beings. As agriculture uses 75% of freshwater resources, demand for water and plants is growing day by day as our population increases. Various approaches have been developed to conserve water in different ways. In conventional irrigation schemes, we need an operator or farmer to carry water to plants; but he does not understand what plant needs how much water to get the correct yield. The notion of smart agriculture is evolving since IOT sensors can provide farming information and then use the feedback of the consumer to act. A smart farming device will be developed in this paper with advantages from advanced skills such as IOT, Arduino and the wireless sensor network. IoT Robotics and Smart Agriculture [2]. Environmental control is the key consideration for improving the performance of productive plants. This document includes the development of a temperature, humidity and even animal motion monitoring scheme that can damage plants in agricultural fields via sensors using the Arduino board and send an SMS and application format notification to the farmer via Wi-Fi/3G-4G on his/her smartphone in the event of discrepancies.

Multiple characteristics have been proposed by Suma et al. [3], including remote, GPS-based monitoring of humidity and temperature, scaring intruders, protection,

leaf humidity, and adequate irrigation equipment. In order to continuously record soil characteristics and environmental variables, wireless sensor networks are used. As before, farmers manually tested the parameters. Rehena et al. [4] defined the specific degree of soil dampness and temperature value and was based on pre-determined estimates of soil humidity and temperature limit. The Arduino board controls high voltage-growing system types without human intercession. Rao et al. [5] concluded that the Internet of Things (IoT) allows plant growth tracking and choice, decision-making support for irrigation, etc. A Raspberry-Pi-based automated irrigation IoT system is proposed to modernize and increase plant production.

The main purpose of this proposed work is to grow plants with less water usage, so that most farmers spend a lot of time in the fields to concentrate on the water that plants have at the time they need. Water control should be improved, and device circuit complexity should be minimized. The proposed system was constructed using sensor data, and the amount of water required was determined. As farmers have no idea how much water should be applied and how plant surveillance can be carried out, there is no idea. The vast amounts of water and the crops' production are not sufficient. Smart farming, which involves modernizing current traditional farming techniques, is the only solution to this issue. This is why automation and IoT strategies for intelligent agriculture are intended to be used.

2 Proposed Methodology

As agriculture is one of the significant factors in the life of the human being to survive. More than 75% of the freshwater resources are used by agriculture, but our population growth is growing day-by-day, with more demand for water and more demand for plants. Many techniques have been developed to conserve water in various ways. We need a farmer to put water on plants in conventional irrigation systems, but often he will not get to know at what time he has to come to store and supply the plants with water. So more effort from the farmers is needed for this manual operation. If the farmer is not going to field at the right time or if he misses going to the field every day, on that day, the plant may be in need of water. So, to stop certain errors or the farmer's manual work. Often farmers manually supply more or less water, which in turn contributes to the unhealthy growth of the plant as well as wastage of water.

With the automation system, we will know about the humidity and temperature on that particular day. This suggested method allows the farmer to get to know the moisture content in the soil for the purpose of minimizing water wastage. From the survey [5], it is clear that the root has the potential to absorb up to 90 cm of water resource. So, 90 cm is divided into three equal parts of 30 cm each, and we will keep the soil moisture sensor at 30 cm each, so that at three different levels the farmer will get to know the moisture content in the soil. Three water pumps and three soil moisture sensors are placed near the plant. Based on the content of the moisture present in the soil, the water pump will be activated, and water will travel to that stage of the soil. The flowchart of the system proposed is shown in Fig. 1.



Fig. 1 Flowchart of the system proposed

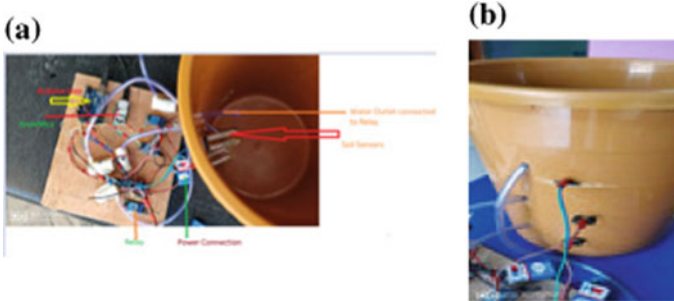


Fig. 2 a Setup depicting the Arduino board Sensor and node MCU connections (top view). b Setup depicting of Soil Relay pump connection

In addition, the soil moisture which will determine the amount of water in the soil must be regularly and according to moisture value, water will be supplied to plants. In the negative scenario if this didn't happen and farmer supply more water to the plant, then plant will not get proper amount of time and space for respiration and growth will not be happening properly and finally plant may die. Hence, as to decrease the water wastage, automation is made where the water is utilized properly. The soil moisture sensor is used to get the measurement of moisture to achieve this. Accordingly, water pumps will get activated. As shown in Fig. 2a, b the real-time implementation is represented using an experimental setup.

The sensor readings are taken, and water is supplied, all the process are done in technical way. So, the android application is set up to transmit to the farmers, which

will display the estimate of the moisture sensor and the water stream of the pump. The conditions are given in such a way that water should be supplied to that layer if the moisture value is below 50%. Here (0–1000), as a percentage considered. The water is not wasted. The required amount of water should be supplied with regard to the amount of moisture present in the soil in order to grow a healthy plant. If this is done correctly, then a healthy plant can grow. That is done by the following method to keep track of the plant, whether it is healthy or diseased [6]. Let us further understand in detail about disease detection in banana plant.

2.1 Disease Detection in Banana Plant

In order to classify a plant as healthy or sick, the disease detection algorithm integrates information from plant thermal, depth and visible light images and uses the classification of features obtained from these images. Here we are applying this disease analysis technique for banana leaf. We identify whether the leaf of the banana is healthy or diseased [7]. The leaf is said to be a diseased leaf by visualizing the infected regions on the leaf. The proposed system of disease detection in plant is shown in Fig. 1. Every part of the banana plant which includes banana fruit, banana flower, banana stem, raw banana, banana leaf has health benefits. The disease detection for the plants [8] is carried out by using Raspberry Pi to capture the picture.

3 Result and Discussion

In the tradition way, farmer must go to field to monitor the growth of the plant and water them properly. But this is hectic to farmer, and certain time farmer will not get to know what should be done for the better way to grow the plant. In this case, technology will help farmer to know better about the plant. In this paper, we are using IoT sensor to help the farmer. The advantage of using smart farming is to know the water level indication to the farmer, and we can get to know the health of the plant using leaf. For smart farming using IoT, we are using soil moisture sensor which will get to know the moisture content in the soil. We are using relay which will be connected to water pump, depending on the values of soil sensor water pump will be triggered automatically. The experiment set-up is show in Fig. 3.

Here we are using soil moisture sensor which will help to get the moisture content in the soil with which we will get to know, how much amount of water is required for that particular area of the soil. By the help of pump, which is connected to the water supplier, will supply exact amount of water which is required for soil.

In Fig. 4, the pot is divided into three levels: the first layer, the second layer and the third layer. It shows the empty pot with three sensors and three water pump connections.

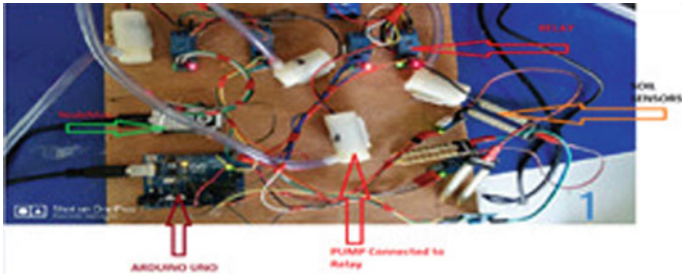
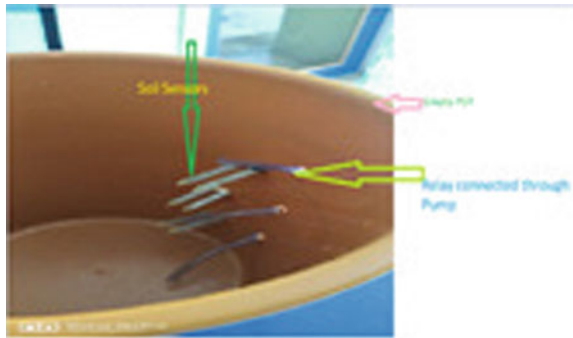


Fig. 3 Experiment set-up of smart farming

Fig. 4 Empty pot with soil sensor and relay pump



3.1 Water Flow Rate

As we know farmer will have to check on daily basis whether sufficient amount of water is supplied or not, whether any nutrition has to be given to the plant for its healthy growth. Farmer will do whatever it takes to do to grow plant, because this is the daily routine of the farmer, through which he will get his bread and butter for his family. To those customers who stay in urban and rural area, they are also able to get food because of hard work of farmer and all are happy.

To help farmer, here we have come up with smart agriculture using IoT, where farmer can sit in home and monitor the growth of plant. We have considered some scenarios based on soil moisture content (i.e. no moisture, moisture only in bottom layer, moisture in only top layer and all three layers has moisture) which tells about the advantages of using IoT in agriculture.

Let us understand each scenarios in detail:

Scenario 1: No moisture content in the soil and dry in all three (i.e. top, middle and bottom) layers

Consider a scenario where it is hot sunny day and the land will totally dry, then in manual case farmer has to go to field and ON the water pump. But farmer will not get to know when to stop the water supply. So here we have a solution for the farmer. Consider the pot where we have divided the pot into three layers, in the same way

Table 1 When land is dry and only top layer is water is supplied

S. No.	S1	S2	S3	W1	W2	W3
1	974	1013	1016	1	0	0
2	983	1013	1016	1	0	0
3	989	1013	1016	1	0	0
4	990	1013	1016	1	0	0
5	991	1013	1016	1	0	0

we have separated the water supply using three water pumps which are named as W1, W2 and W3 and soil sensor as S1, S2 and S3. With the help of soil sensor, we will get to know the value of moisture content in the soil. By that water pump will be triggered to top layer of the soil, and we are doing this because to control the wastage of the water. Once top layer gets sufficient amount of water, then water supply stops. So that middle layer water pump will be triggered, and water will be supplied.

From the Table 1 we can see, soil sensors are dry, i.e. S1, S2 and S3 are more than 50% (500) and W1 relay is ON as per the calculation, and we can observe it in Fig. 5a 1st relay is on with green light. For Example, in row number 1—S1, S2, S3 are more than 500 and W1 is 1 (i.e. ON) and W2, W3 are 0 (i.e. OFF).

Scenario 2: Moisture content in the soil is present only in the bottom layer

Consider another scenario where bottom layer is moisture and top two layers are dry. This scenario occurs when it has rained heavily, or farmer has supplied more amount of water to field. After two days of heavy rain, the top layer will be dry, but bottom layer will still have moisture content in the soil; however, this will not be visible to farmer, so he will start to supply water to field. If farmer does this, then bottom layer will have more amount of water which is access with the capacity of plant, if this happens again and again, plants starts to spoil. But this will not be visible to farmer and he will be dependent on the outcome of that plant. But plants would have start to spoil and after some days it will die.

To solve this issue, we have soil sensor and water pump through which we will get to know the moisture content in the soil and depending on that water pump will be triggered. So, water will be supplied to the plant when it is required, and sufficient amount of water not access to that. Here we go with technical term. If S1 and S2 are dry (meaning 1st, 2nd layer are dry and 3rd layer has moisture) than W2 will be ON (meaning 2nd relay will be triggered), we can observe that in Fig. 5b. In Fig. 5b, we can observe soil sensor which is in water is 3rd sensor, and 2nd relay is ON with green light On.

In Table 2, soil sensors are dry, i.e. S1, S2 are more than 50% (500) and S3 is less than 50%, i.e. it has moisture content in the soil and W2 relay is ON as per the calculation, and we can observe it in Fig. 5b and 2nd relay is ON with green light. For example, in S. No. 1 S1, S2 are more than 500 (i.e. it doesn't has moisture) and S3 is less than 50%. So, W2 water pump will be ON and W1, W3 are 0 (i.e. OFF).

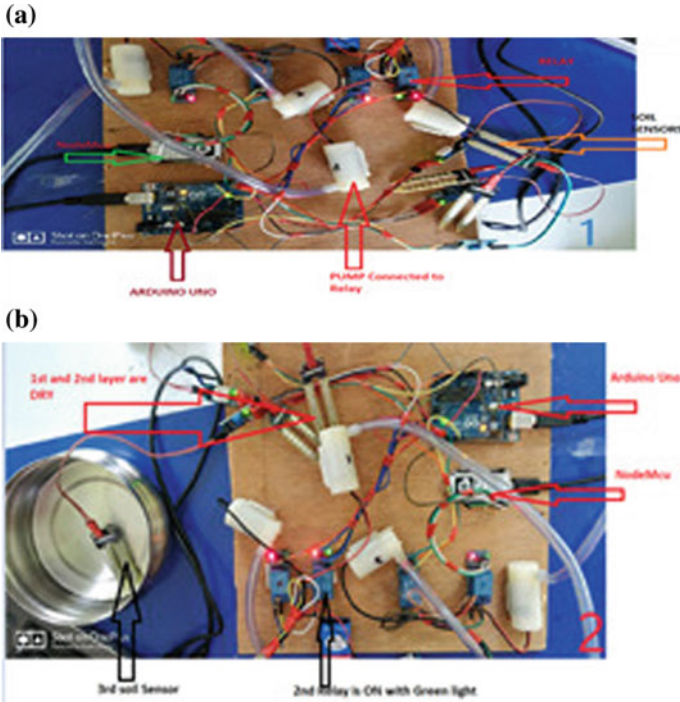


Fig. 5 a No moisture—water is supplied to top layer through 1st relay which is ON (i.e. green light). b Top two layers are dry and 3rd sensor has moisture, second relay is ON

Table 2 1st and 2nd layers of soil are dry and 3rd layer has moisture, so second relay water pump will be ON

S. No.	S1	S2	S3	W1	W2	W3
1	1011	1012	254	0	1	0
2	987	1012	287	0	1	0
3	964	1011	264	0	1	0
4	1012	1011	290	0	1	0
5	1011	1012	254	0	1	0

From all the scenario, we have come to know that, from the help of soil sensor and water pump relay, the moisture content in the soil and required amount of water is supplied to soil at that moment automatically by the help of water pump. Hence from Tables 1 and 2, we can see the value of soil sensor and which water pump is triggered. So by this experimental set-up, we have explain that we have reduced the wastage of water and get to know when to supply the water to plant and how much amount of water is required and hence, reduced the farmer effort and human error as well.

3.2 *Banana Leaf Monitoring*

Now farmer will obviously focus on the growth of the plant; here we are checking the health of the leaf to monitor the growth. So, we will get to know whether plant is growing properly with time to time [10–12]. In this paper, we have taken banana plant as this plant is useful in many ways and it is used in every season of the years. It is having most advantage with the leaf and as well as with the fruit and every part of this plant is having health benefit. Once farmer starts to water the plant, he will start to expect the growth of the plant in such a way that he will benefited. But from the outer visibility, he will not get to know whether plant is growing properly or not. So that farmer might put more pesticides, and if it is excess that is also harm and he might supply more amount of water that will also cause harm to plant. So, if this type of scenario occurs, he will not get best output and farmer will be sad. We have found one of the way where farmer can monitor it closely and grow the plant. We have taken the banana leaf as an example. Here we are checking whether the leaf is healthy or diseased, by which we can determine whether the plant is growing properly or not. With the help of Raspberry Pi, we will take the pic of the healthy or diseased leaf and store as a data and we use SVM model to get the result.

By this we will get to know whether the plant is growing properly or not. As a future work if the leaf is non-healthy leaf, then we can spray a pesticide to the leaf to grow properly (Figs. 6 and 7).

3.3 *Android Application*

All these technical problems are not explicitly known to the farmer, such as sensor values, Wi-Fi connection and client-server transformations. As our nation is moving towards digitalization and farmers also use mobile, an android application is built for this purpose from which he can get to know about the value of soil moisture and pump that is active. The android application interface is shown in Fig. 8 which gives

Fig. 6 Raspberry Pi taking pic of healthy banana leaf

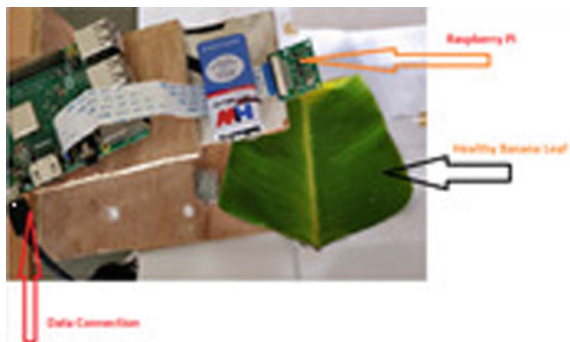


Fig. 7 Raspberry Pi taking pic of diseased banana leaf

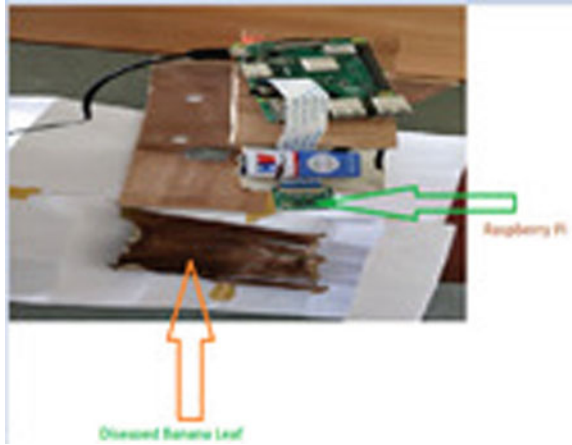


Fig. 8 Android application interface showing sensor and pump values



the information regarding three soil sensor values and three water pump values. As we check the health of the plant by monitoring the leaf, those picture will also be seen in android app. If it is healthy leaf, then it looks as in Fig. 9 and if it is diseased or unhealthy leaf it will be seen as in Fig. 10.

4 Conclusion

Agriculture is the industry that consumes more water, accounting for 75% of global water use. But as the population and food demand is rising, there is also a great demand in water and plants. As we have discussed in this paper, in traditional farming

Fig. 9 Android application interface showing the healthy banana leaf

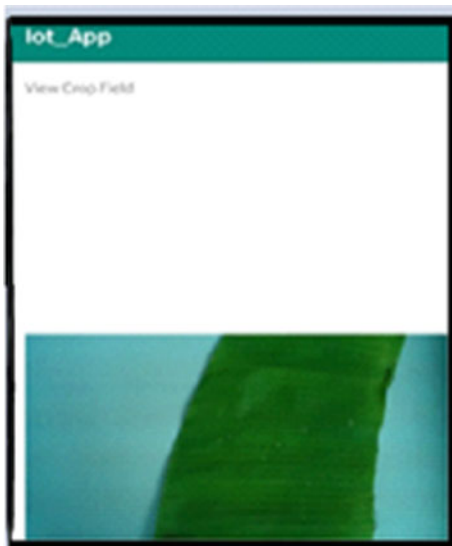


Fig. 10 Android application interface showing the diseased banana leaf



system, it was difficult for the farmer to estimate the amount of water required to the plant, because of which farmer was supplying more water to the plant which is more than the consumption of plant or less amount of water where plant will not grow properly. The proposed model for the farming sector is basically a careful estimation of the water which is required by the soil which will help to minimize the water wastage. So, we are using smart farming system which will help farmer to get to know the moisture content in the soil, by which water relay will be automatically triggered to supply water to plants. As shown in result section, this is achieved by dividing the soil absorption potential into three equal quantities with the help of the soil moisture sensor and the water pump, so that it estimates the moisture content in the soil, concerning that water will flow to that layer which has less moisture. By this experiment, we have achieved to reduce the water wastage. Once wastage of water is taken care, the next step is to check the health of the plant, which is done by monitoring the leaf of banana plant. We were able to achieve this by using SVM algorithm. This is used to segregate the leaf as healthy or diseased leaf. By this work, we are able to save the water and also check the health of the plant at the early stage.

References

1. Sushanth G, Sujatha S (2018) IOT based smart agriculture system. In: International conference on wireless communications, signal processing and networking, Chennai, pp 1–4
2. Dholu M, Ghodinde KA (2018) Internet of Things (IoT) for precision agriculture application. In: 2018 2nd international conference on trends in electronics and informatics (ICOEI), Tirunelveli
3. Madhav A, Bhamini NM, Suma HN (2021) An IoT based intravenous drip rate controlling and monitoring device. In: 2021 international conference on COMMunication Systems & NETWORKS (COMSNETS), Bangalore
4. AshifuddinMondal M, Rehena Z (2018) IoT based intelligent agriculture field monitoring system. In: 2018 8th international conference on cloud computing, data science & engineering (confluence), Noida
5. Rao RN, Sridhar B (2018) IoT based smart crop-field monitoring and automation irrigation system. In: 2nd international conference on inventive systems and control, pp 478–483
6. Sharath DM, Akhilesh, Kumar SA, Rohan MG, Prathap C (2019) Image based plant disease detection in pomegranate plant for bacterial blight. In: 2019 international conference on communication and signal processing (ICCSP), Chennai
7. Song C, Wang C, Yang Y (2020) Automatic detection and image recognition of precision agriculture for citrus diseases. In: 2020 IEEE Eurasia conference on IOT, communication and engineering (ECICE), Yunlin
8. Devaraj A, Rathan K, Jaahnavi S, Indira K (2019) Identification of plant disease using image processing technique. In: 2019 international conference on communication and signal processing (ICCSP), Chennai
9. Kumar SS, Raghavendra BK (2019) Diseases detection of various plant leaf using image processing techniques: a review. In: 2019 5th international conference on advanced computing & communication systems (ICACCS), Coimbatore
10. Hossain K, Rahman M, Roy S (2019) IoT data compression and optimization techniques in cloud storage: current prospects and future directions. *Int J Cloud Appl Comput (IJCAC)* 9(2):43–59
11. Singh N, Vardhan M (2019) Distributed ledger technology-based property transaction system with support for IoT devices. *Int J Cloud Appl Comput (IJCAC)* 9(2):60–78

12. Kumar SS, Raghavendra BK (2019) Diseases detection of various plant leaf using image processing techniques: a review. In: 2019 5th international conference on advanced computing & communication systems (ICACCS), Coimbatore

Securing the Smart Devices in Home Automation System



Syeda Sabah Sultana and J. Sangeetha

Abstract Security is the major concern in every infrastructure such as offices, banks, hospitals, etc. Due to lack of security in the existing home automation system, hackers can easily access and collapse the system. Hence, in this research work, we are providing security to the remotely controlled home infrastructure and to reduce the energy consumption of smart devices. The smart device consists of the appliances such as four lights and two switches operated using the web-based application. This application allows only authorized users to remove these smart devices which are not in use for longer time and also the hacked devices. Thus, we are reducing energy consumption and securing our smart devices from hackers. Adding to the benefit of the user, through this application we can change the status of the devices either to on or off state. The status of the devices is stored in the cloud in the encrypted form (ciphertext) using the encryption technique such as the AES algorithm. Through the server using Wi-Fi module, this ciphertext can be accessed by the user credentials and decrypt it through web-based application. In this research work, major security issues like authentication and verification are taken care, and this work focus is on reduction of energy consumption by removing unnecessary devices and protecting the smart devices from hackers in the existing home automation system.

Keywords Internet of Things · AES algorithm · Energy consumption · Verification · Authentication

1 Introduction

The Internet of Things interconnects digital devices across Internet into everyday devices that enable them to send and receive data. It plays an important role in our daily lives. The Internet of Things contributes too many areas such as education,

S. S. Sultana (✉) · J. Sangeetha
Department of Computer Science and Engineering, M S Ramaiah Institute of Technology,
Bengaluru 560054, India
e-mail: syedasabah.01@gmail.com

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
D. P. Agrawal et al. (eds.), *Cyber Security, Privacy and Networking*, Lecture Notes
in Networks and Systems 370, https://doi.org/10.1007/978-981-16-8664-1_24

273

health system, automobiles, entertainment, smart home, etc. It has many challenges such as security threats, data leakage, data manipulation, and other vulnerabilities [1].

In [2], the approach to machine learning (ML) [2] is showed that accelerometer data is used to trade in with problems with Gesture Recognition (GR). Objectives of the approach are to provide classification with high accuracy that are typically independent of the user and devices, independent and system orientation for home automation systems, a heterogeneous scenario in earlier GR literature; this was not thoroughly explored.

In [3], simplistic simulation of a full home automation system has been carried out in using components and raw materials that are readily available. The system automates five separate devices/loads that can be accessed via an Internet connection through a webpage from anywhere in the world. With a login function, the webpage thus given is protected. A revolutionary utilization monitoring feature, which monitors the length of use of and system after each ON-OFF cycle, has been established. It allows consumer to monitor and reduce use, energy utilization and therefore effectiveness bills strategically.

In [4], it present a new intelligent smart home concept that incorporates the IoT concept based on a web application. In addition, a communication model is specified for exchanging data in the same medium. It provides a common medium for all heterogeneous devices to communicate. In addition, based on web applications, device architecture is also proposed. To send or receive action messages over the network, the web application concept is used. The architecture presented offers the aspects of implementation, study and visualization in which different devices interact with other devices. Similarly, energy usage is also computed for the sensors installed in the proposed smart home. The energy consumption of the sensors using the architecture proposed is substantially less. The final assessments of the network architecture meet the user-related needs, whether the input data is real time or offline when taking real-time action.

The author [5] introduces smart home and security system scheme, also in brief present the system's architecture, system's functionality, interface of system, identity addressing and security mechanism. In this system, sensing and home gateways allow connecting network of all levels allowing users to query data at any instance, controlling of devices on home network. The system's performance is measured. As sensing technologies advance, more study focuses on increasing the efficiency of the system and security framework of the system to satisfy the requirements of users' privacy.

In [6], in order to control switch-based appliances through human speech, there are human perception problems that comes from human speech. Also the missing of necessary parameters is a problem to understand for identifying an object by a computer. To overcome the inherent issues, context information is employed. Through this, it can provide indication to understand human speech to be used as a control command for home appliances. Thus the paper concludes with the monitoring of proposed control system by sensors. Context information previously inferred or a user at home can monitor the system.

In [7], the existing network infrastructure IoT allows devices to remotely control. IoT allows these devices to be incorporated into computer-based systems, resulting in increased performance, precision and cost savings while requiring less human interaction. Cyber-attacks harm other devices if they are secured poorly, and they use them as gateway which results in security and privacy issues in the network. The author focuses more on the constraints and security challenges. It concentrates on the challenges gained by the IoT-connected devices as well as their capability to connect. Also communication between the devices and remotely managing of large number of automated devices via the Internet.

In [8], numerous elements are contained in an IoT solution that effects the execution of security and privacy features which brings a functionality concern. Some elements such as open-source and proprietary are also among those that users are unable to control them. However, a smart home device can be controlled by user via application of smart home remotely. This consists of embedded devices linked to the cloud. To grant digital entity, a lightweight identity stack is proposed for IoT with the devices and users that interact with them. An authentication scheme is used for Fast Identity Online (FIDO). Every time a FIDO authentication receives request from the user, a keep-alive protocol is used.

In [9], there are many key challenges in IoT objects, IoT objects need simple security solutions that mostly function at minimum energy levels and with lesser amount of capabilities thus resulting in hindrance of difficult security solutions. This hindrance is caused due their memory and computational requirements such as cryptographic protocols. One way of securing devices against emulation attacks is the use of environmental-based fingerprinting. To authenticate devices, device fingerprinting is a technique that uses unique features extracted from the objects transmitted signals and environment.

In [10], the author focused on an authentication scheme on IoT devices. To support the authentication, all set of devices communicates with a gateway. There is connection between controller and the gateway which can access to the central data. The access can be provided by passing the authentication scheme through gateway and controller. There are three levels where the message flows between: things, gateway and the controller. The first phase requires obtaining a public key certificate by a gateway through the controller. The second phase starts by thing by sending an authentication request to the gateway. The last stage is requisition of authentication from IoT device to gateway. Testing is completed by the tool AVISPA. Evaluation of result shows the identity-based authentication scheme is opposed to various attacks.

The author [11] explains that RFID is vulnerable to attacks on security and privacy. This is because any request via wireless communication from a reader, the RFID responds to its unique ID. Due to the non-selective response of RFID tags to all reader queries, the items recognized with tags may reveal information that is insightful. Through this, the adversary can attain trace goals rely. Physical attacks, cheat tags, DoS attacks, eavesdropping and communication flow analysis and other security issues are all faced by RFID systems. Tag reading by attackers can be executed without suitable control solutions. Thus, the functionality of each application has been more focused by users.

For end-user appliances, the Home Energy Management System (HEMS) is presented through hardware demonstration [12]. The contact delay time of the HEMS used to perform load control and consumption of energy are investigated in detail.

The objective of this research work is to focus on the challenges of IoT concerning security and privacy. The main purpose of this work is to provide security to the devices in the existing smart home system and to reduce the consumption of energy used by these devices. The system consists of the appliances such as four lights and two switches. It is operated using web-based application. The application allows removing the devices which are not in use for longer time in order to reduce energy consumption and to protect the devices which are hacked by the hackers. It also allow to change the status of devices to on or off state. The status of the devices is stored in the cloud in the encrypted form. An AES algorithm of 128 bit is used as the encryption technique that generates the ciphertext. Through the server using Wi-Fi module, this ciphertext can be accessed by the user credentials and decrypt it through web-based application. In this research work, major security issues like authentication and verification are taken care and this work focus is on reduction of energy consumption by removing unnecessary devices and protecting the smart devices from hackers in the existing home automation system.

The system provides security to the existing system in two levels: In the first level, the credentials are verified by the server that comes from the user for the authentication, and the second level is to provide verification through the generated OTP (One Time Password) and notifies the user through an e-mail in case of animosities such as manipulation of data or the hacker trying to hack the data. This paper aims to shed light concerning security requirements: authentication, confidentiality in existing home automation system. Further, the organization of the paper is as follows: Sect. 2 explains the methodology of the proposed system, Sect. 3 gives the exemplar of AES algorithm, and Sect. 4 provides results of the proposed model and finally the conclusion is discussed.

2 Proposed Methodology

In this research work, we have considered the following hardware module: Arduino UNO, Raspberry Pi, Wi-Fi module (Node MCU) and Relay module. The hardware design in our existing automation system consists of devices (i.e., four lights and two switches). For the better understanding and security of these devices, they are identified with device identifiers (IDs) with the naming convention as in Table 1.

The connection between client and server is formed through the activated Wi-Fi option available in the smartphone. Raspberry Pi acts as a server side. The Internet Protocol (IP) address is configured to the Virtual Network Computing (VNC) viewer to connect to the Raspberry Pi. The Arduino Uno board consisting of digital pins is connected to each IoT device in the system. Each system is connected via relay to the arduino. A Python program loaded onto the Arduino Uno board's microprocessor

Table 1 Device name and identification of the devices

S. No.	DeviceName	DeviceID
1	Light1	SS01
2	Light2	SS02
3	Light3	SS03
4	Light4	SS04
5	Switch1	SS05
6	Switch2	SS06

chip helps to perform action when particular input is received. End-user web-based application is used to monitor and control the smart devices from any remote location.

The software module consists of the web-based application and cloud solution. A web-based application provides user a user interface. Every user needs to register with username and password. The application provides two levels of securities. In the first level of security, when IoT devices are accessed by the user, it sends requests to the server using their login credentials and this is authenticated by the server. These credentials are stored in the server. For authentication, the server verifies the information provided by the user such as email address, password. When the attacker makes many failed login attempts, i.e., up to three attempts, an email is sent to the real user requesting that they change their password immediately. In the second level of security, the user's identity will be verified through the use of an OTP. The user can then modify the status as on or off state of the devices via the cloud. Here the status of device 'on' is considered as 'yes' and 'off' as 'no'. The application can remove the devices once the two-level securities have been successfully completed. It allows you to periodically update your profile and change your password. This application is used to modify a device's status from its prior state. It is encrypted and saved in the cloud as ciphertext. As a result, the ciphertext produced uses the AES algorithm which is an encryption technique.

3 AES Algorithm Explanation

The encryption technique is implemented using AES algorithm. The AES algorithm is a symmetric-key cipher that encrypts the data using a single key shared by both sender and recipient. The AES algorithm here uses the Shift-Row Transformation technique to generate the ciphertext.

Let us understand with an illustrative example, where the plaintext is 'Yes'. Converting the plaintext into Hexadecimal form, we have the values for 'Y' as 59, 'e' as 65, 's' as 73 and for 'whitespace' as 20. Hence considering the entire plaintext with space up to 16 bytes, the hexadecimal value for the plaintext is

59 65 73 20 20 20 20 20 20 20 20 20 20 20 20

Consider the key Dd_WFBROXfRbaHUX for round 0 key.

Converting the key into hexadecimal form, we have the values as: for 'D' the hexadecimal value is 44, for 'd' is 64, '_' is 5F and so on the values goes on for the rest of the bytes in the key.

The final hexadecimal value for the key is:

44 64 5F 57 46 42 52 4F 58 66 52 62 61 48 55 58

Key expansion process is used by AES algorithm to create round keys for each key. It is created word by word in an array form. Each word consists of 4 bytes. The following are the steps to be followed:

Step 1: First four words (w_0 , w_1 , w_2 , w_3) are generated from the key; each word consists of bytes $w[0] = (k_0, k_1, k_2, k_3)$, $w[1] = [k_4, k_5, k_6, k_7]$, $w[2] = [k_8, k_9, k_{10}, k_{11}]$ and $w[3] = [k_{12}, k_{13}, k_{14}, k_{15}]$.

Here in our algorithm, the following are the words consisting of 4 bytes.

$w[0] = (44\ 64\ 5F\ 57)$ $w[1] = (46\ 42\ 52\ 4F)$

$w[2] = (58\ 66\ 52\ 62)$ $w[3] = (61\ 48\ 55\ 58)$

Copying the last four bytes of the existing key to a four-byte temporary vector

$w[3] = (61\ 48\ 55\ 58)$

Step 2: Circular byte left shift

It takes a word $w[3]$ of 4 bytes and perform shift operation each byte to the left as shown below

$w[3] : (48\ 55\ 58\ 61)$

Step 3: Byte Substitution (S-Box)

This step relies on nonlinear S-Box. In this step, a byte in the state is replaced to another byte which is called as Rijndael S-box.

(S-Box) for $w[3] : (52\ fc\ 6a\ ef)$

Step 4: Adding round constant RCON

Each RCON is a four-byte value, where the rightmost 3 bytes are always 0 where

$RCON[i] = [x \wedge i - 1, 00, 00, 00]$

The values $x \wedge i - 1$ are to be computed in the same representation of Galois field (GF)

$$(GF) = (2 \wedge 8)$$

Since we are calculating key for round 1, we need the RCON value as RCON[1] = [01, 00, 00, 00]

$$g(w[3]) = RCON[1] \text{ XOR } w[3]$$

$$g(w[3]) = (01\ 00\ 00\ 00) \text{ XOR } (52\ fc\ 6a\ ef)$$

$$g(w[3]) = (53\ fc\ 6a\ ef)$$

$$w[4] = w[0] \text{ XOR } g(w[3])$$

$$w[4] = (44\ 64\ 55F\ 57) \text{ XOR } (53\ fc\ 6a\ ef)$$

$$w[4] = 17\ 98\ 35\ b8$$

$$w[5] = w[4] \text{ XOR } w[1]$$

$$w[5] = 51da\ 67\ f7$$

$$w[6] = w[5] \text{ XOR } w[2]$$

$$w[6] = 9\ bc\ 35\ 95$$

$$w[7] = w[6] \text{ XOR } w[3]$$

$$w[7] = 68\ f4\ 60\ cd$$

The first round key generated is (17 98 35 b8 51 da 67 f7 9 bc 35 95 68 f4 60 cd). Similarly, we can generate for the remaining nine rounds using AES algorithm.

There are lots of attacks done by the eavesdroppers. One of the popular attacks is the brute-force attack, a method of trial and error in order to get the original data. The AES algorithm is computationally secure than Data Encryption Standard (DES) against this attack because it is not possible to acquire 128 bit key to get attacked.

4 Results Analysis

In this research work, two levels of security are implemented: authentication/authorization of the system and verification/validation of the system.

The first level of security, which is especially important from a system-wide perspective, is the authentication and user authorization. The result analyses are explained in detail.

In the first level of security, the user must first register with the Smart Home Automation System by entering his or her information such as Name, Email-ID, Phone Number and Password on the Sign-In page. The user is authorized to input the password that is saved in the server once the Email ID has been registered. When a user logs in using his or her registered credentials (username and password), the server verifies the user identification whenever the user logs in with the registered credentials, i.e., username and password. If the entered password is incorrect three times in a row, a login alert is sent to the registered Email-ID, requesting that the password be changed immediately. This assures that the user's credentials are secure and that no attacker may mislead the information and uses it for malicious attacking the system. As a result, the IoT devices are safe and inaccessible to attackers. Thus, the system allows the user to change their password to a new one. Passwords should be changed on a frequent basis to keep the system secure. The system has a functionality that allows you to update your profile. This feature allows you to modify your profile and password. Users can update their profile by changing their name and phone number.

In the second level of security, the server verifies the user's identity by requiring the user to enter a four-digit One Time Password (OTP). An OTP is sent to the user's registered email address. The server allows access to the system based on the user's identification. After a successful login, a web page providing complete control of the system is displayed.

From Figs. 1 and 2, we can observe that the user uses the system to turn the lights and switches such as Light1, Light2, Light4, Switch1 in on state and to turn the light and switch such as Light3 and Switch2 in off state.

4.1 *Removing of Devices*

The system provides an additional feature of removing devices from the system. The user has the flexibility of removing devices when they are no longer required by



Fig. 1 User interface for operating the devices

Fig. 2 Output status of devices (i.e., Light, Light2, Light, Light4, Switch1, Switch2)



Fig. 3 Removing of device successfully

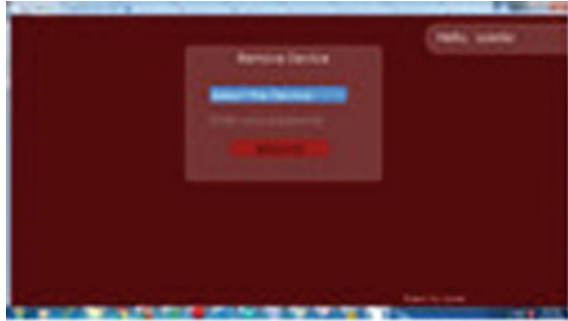


Fig. 4 After successfully removing of device



selecting the device and entering the correct password as shown in Fig. 3. The alert message is shown on removing of device successfully.

Assume we have 3 devices Light1, Light2, Switch1, we can observe that device “Switch1” is removed successfully from the application as shown in Fig. 4.

On successful removing of device, an email is sent to the registered user’s Email-Id as shown in Fig. 5.

Also, it is necessary to remove unnecessary devices that are no longer used as this will consume energy. The device can be removed using the remove device option from the system. This helps reduce energy consumption and also care is taken, wherein



Fig. 5 Email on removing of device

only concerned person will be allowed to remove the devices. In this way, two-layer security is provided, in which the first level of security requires user to enter correct password for removing the devices and the second level of security which notifies the user through email whenever the device is removed from the application.

There are also larger chances of the devices getting hacked by the hacker, and the removing of device functionality provides security to the system by removing the device.

4.2 *Cloud-Based Solution*

In this research work, a secure cloud-based is employed as the solution. It allows IoT devices to connect and communicate with one another. Because of the limited resources available in IoT, a substantial amount of data generated by IoT devices must be communicated to the cloud, and all devices must be accessible through the Internet. Because users have little control over their cloud services, they must trust their cloud providers to provide adequate security methods for their data. As a result, encryption is used in this approach. As a result, the encryption utilized in this approach ensures that end-to-end communications are secure.

For instance, in the existing system, the input data/plaintext we considered were the on or off status of devices. The statuses 'on' and 'off' have been interpreted as 'yes' and 'no,' respectively. This information is obtained from the Arduino board and is used in the encryption process.

Consider Device ID SS01, the input data/plaintext is the status of devices on or off. This information is received from the Arduino board. Let AC be the state of device, Y is called the active, and KY represents the key which is being used to perform addition of matrix.

In this research work, 128-bit key size is utilized. The key calls for ten rounds. The number of rounds is determined by the key size utilized. The ciphertext generated for the individual device ID in the cloud is represented by CT. We may also produce ciphertext for other devices using their unique device IDs.

The user can access the cloud to change the status (ON/OFF) of the devices in this existing system. For each device ID such as SS01, SS02, SS03, SS04, SS05, SS06, the algorithm generates a unique ciphertext which is stored in the cloud whenever the user changes the device status from on to off or vice versa. As a result, the confidentiality between sender and receiver is preserved.

Furthermore, data encryption is essential, because we are using the open source Firebase cloud, an attacker can obtain the device's system information without the user's consent or awareness. These aid attackers in deciphering patterns of user behavior in order to track devices. Because an attacker can remotely change the status of the devices, posing a threat to the user's devices.

Assume we're accessing and changing the state of devices via a public network outside of our home, such as Wi-Fi in a mall, train station or other public location.

An attacker with access to data stored in the cloud can intercept this communication between the devices and the cloud. As a result, data saved in the cloud and IoT devices are secure, and no attacker can access them.

5 Conclusion

In this paper, we have discussed security and energy consumption as the major problems in the existing smart system. If any device is hacked by the attacker, it will provide the user with an alert mail on their respective credentials which can protect the system. The AES algorithm aids in the generation of cipher text, which is used to securely store data in the cloud. It also ensures that the cloud is secure, where the status of the devices is stored in encrypted form so that no attacker should be able to change the state of the device. The system also focuses on the second major problem of reduction of energy consumption by the devices. The functionality of removing device provides user to remove unnecessary devices which are not in use for longer time and the hacked devices. Thus, reducing the energy consumption and securing the smart devices from the hacker.

Acknowledgements We would like to thank M.S Ramaiah Institute of Technology, Computer Science and Engineering for supporting this research work.

References

1. Granjal J, Monteiro E, Sá Silva JS (2010) A secure interconnection model for IPv6 enabled wireless sensor networks. In: IFIP wireless days, Venice, pp 1–6
2. Cenedese A, Susto GA, Belgioioso G, Cirillo GI, Fraccaroli F (2015) Home automation oriented gesture classification from inertial measurements. *IEEE Trans Autom Sci Eng* 12(4):1200–1210. <https://doi.org/10.1109/TASE.2015.2473659>
3. Paul S, Indragandhi V, Kumar NK, Raja Singh R, Subramaniaswamy V (2019) An IoT based home automation system. *IOP Conf Ser Mater Sci Eng* 623:012014. <https://doi.org/10.1088/1757-899x/623/1/012014>
4. Khan M, Din S, Jabbar S, Gohar M, Ghayvat H, Mukhopadhyay SC (2016) Context-aware low power intelligent smart home based on the Internet of Things. *Comput Electr Eng* 52:208–222. ISSN 0045-7906
5. Ting J, Yang M, Zhang Y (2012) Research and implementation of M2M smart home and security system. *Secur Commun Netw* 8. <https://doi.org/10.1002/sec.569>
6. Han Y, Hyun J, Jeong T, Yoo J-H, Hong JW-K (2016) A smarhome control system based on context and human speech. In: 18th international conference on advanced communication technology (ICACT)
7. Balamurugan S, Ayyasamy A, Suresh Joseph K (2018) A review on privacy and security challenges in the Internet of Things (IoT) to protect the device and communication networks. *Int J Comput Sci Inf Secur (IJCSIS)* 16(6)
8. Chifor B-C, Bica I, Patriciu V, Pop F (2018) A security authorization scheme for smart home Internet of Things devices. *Future Gener Comput Syst* 740–749

9. Sharaf-Dabbagh Y, Saad W (2016) On the authentication of devices in the Internet of Things. In: IEEE 17th international symposium on a world of wireless, mobile and multimedia networks (WoWMoM), June 2016, pp 1–3
10. Salman O, Abdullah S, Elhadj IH, Chehab A, Kayssi A (2016) Identity-based authentication scheme for the Internet of Things. In: IEEE symposium on computers and communication (ISCC), June 2016, pp 1109–1111
11. Feng H, Fu W (2010) Study of recent development about privacy and security of the Internet of Things. In: International conference on web information systems and mining, Sanya, pp 91–95
12. Kuzlu M, Pipattanasomporn M, Rahman S (2012) Hardware demonstration of a home energy management system for demand response applications. IEEE Trans Smart Grid 3(4):1704–1711. <https://doi.org/10.1109/TSG.2012.2216295>

Dual-Channel Convolutional Recurrent Networks for Session-Based Recommendation



Jingjing Wang, Lap-Kei Lee, and Nga-In Wu

Abstract Recommender systems assist a Web application user in satisfying their needs or interests based on the user profile and past activities. Yet due to privacy and other concerns, some applications and services only keep anonymous information. A session-based recommender system (SRS) predicts the next item by exploring only anonymous user-item behavior orders during ongoing sessions. Recurrent neural networks (RNNs) and their two variants have dominated the research on SRS. However, there are two shortcomings in these RNN-based methods: (1) RNNs easily generate false dependencies because RNNs assume all adjacent items are highly dependent on each other; (2) the sequentially connected architecture of RNNs can only capture the point-level dependencies but ignoring neglecting the union-level dependencies. This paper proposes a Dual-channel Convolutional Recurrent Neural Network (D-CRNN) model to address these problems. This hybrid model leverages RNN to explore complex long-term dependencies and combines CNN to extract the union-level context features, which help to reduce the noise. The hybrid model was evaluated on three commonly used real-world datasets. The experimental results on Diginetica dataset D-CRNN showed an improvement of 5.8% and 4.8% respectively in terms of Recall@10 and MRR@10, demonstrating the effectiveness of D-CRNN on the session-based recommendation.

Keywords Session-based recommendation · Convolutional neural networks · Recurrent neural networks · Hybrid neural networks

J. Wang (✉) · L.-K. Lee

School of Science and Technology, Hong Kong Metropolitan University, Ho Man Tin, Hong Kong SAR, China

e-mail: s1245831@ouhk.edu.hk

L.-K. Lee

e-mail: lkleee@ouhk.edu.hk

N.-I. Wu

College of Professional and Continuing Education, Hong Kong Polytechnic University, Kowloon, Hong Kong SAR, China

e-mail: ngain.wu@cpce-polyu.edu.hk

1 Introduction

Recommender systems (RSs) act an important role in real-world applications by assisting users in satisfying their needs or interests based on the sequential records of user-item interactions, such as rating, viewing, or clicking items. However, in many real-life scenarios, due to privacy and other concerns, informative profiles are not provided, and only anonymous and chronological behavior orders during ongoing sessions are available. Approaches that model these user past interactions on anonymous sessions to predict the next action are called session-based recommender systems (SRS).

The most widely used methods for SRS are Recurrent Neural Networks (RNN) and their two variants, namely the Long Short-Term Memory (LSTM) [1] and the Gated Recurrent Unit (GRU) [2]. RNNs can remember former states to hold the long-term dependencies in the dynamic and evolving sequences, benefiting from the internal memory cells and loop operation. However, gradient vanishing will be easily caused via multiple layer iterations, which motivates the emergence of the two variants. Though these variants solved the gradient issues and obtained exciting results, there are still two shortcomings in these models: (1) The sequentially connected network structure in RNN makes it easily generate false dependencies because not all the adjacent are related; irrelevant and noisy interactions such as clicked out of curiosity or by accident commonly occur in a real-world session. (2) RNNs focus on explore the transition correlation at the point-level while ignore the collective dependencies at the union-level.

For example, given a session $S_1 = \{a \text{ bacon, a rose, eggs, bread, a box of butter, an iPhone}\}$ which denotes items added into the cart successively by a user. The user first purchased food for breakfast, including bacon, egg, bread, and butter, and then added pieces of rose for decoration due to being attracted by some advertisements. Finally, the user ended this shopping session by picking up an iPhone. In this case, the user's purpose has been changed from breakfast to cellphone dynamically. Yet no matter which purpose is, the item "rose" is irrelevant to the whole sequence and is a noisy item. The breakfast can be regarded as the long-term preference, and the cellphone is the short-term preference as it is closer to the prediction one. Then the next item with a high probability is a bottle of milk for breakfast or a phone accessory for the cellphone. Obviously, an effective recommender method should not only encode the sequential items with the long-term and the short-term dependencies but also need to distinguish the noisy items and alleviate their influence. Apart from this, we also observed that in the real world, not all the items needed to be strictly ordered in the sequences, e.g., if the model wants to recommend an airpod, it needs the user to buy an iPhone first; while it makes little sense whether to buy eggs or bread first when a user wants to buy milk for breakfast. As to the considerable sequential patterns, RNN performs well; while others maybe not, such as in the breakfast case, the milk will have a higher probability of purchase when both eggs and bread are already purchased. The above observations occurred commonly in the real world as various advertisements appeared to catch users' attention, and thus any items may be added

to the cart. These phenomena indicated that relying on the pure RNN model might not be the perfect solution for an SRS.

In this paper, we explore a novel hybrid model of combining Convolutional Neural Network (CNN) and RNN to make up for the drawbacks in RNN mentioned above. CNNs are capable of extracting complex local patterns, and the convolution operation is beneficial to filter out the irrelevant features in the current reception field. Furthermore, these local features captured by CNN can be regarded as a union-level correlation between items. Specifically, we propose a hybrid dual-channel modeling architecture with RNN and CNN, named D-CRNN, to capture user behavior in a more reasonable way to fit the shortcoming of using pure RNN-based models. We first treat the item embedding matrix as an image and search the contextual information using CNN. Then the output of the different size convolutional filters will be separately fed into dual-channel RNN to generate the long-term preferences. At last, we apply the target-aware attention mechanism to assign different weights to each channel to generate the session representation.

Our contributions of this work are as follows:

1. A multi-channel D-CRNN utilizes the RNN to explore long-term sequential patterns and leverages CNN to enhance the impact of the short-term dependencies, which is helpful to alleviate the effects of noises.
2. Each channel can capture the long-term and short-term dependencies with different filter sizes, and then adaptively activate channels with the target attentive network to generate the final session representation.
3. We conduct the experiments and analysis on three real-world session datasets. Experimental results demonstrate the effectiveness of D-CRNN.

1.1 Related Work

This section introduces the related work based on pure RNN or CNN and some hybrid models.

RNN-Based SRS. The recurrent architecture of RNN makes it a natural solution for the sequential problems on deep learning-based SRSs. In other words, RNN and its variants (LSTM and GRU) have dominated the studies on SRS. The representative model is proposed by Hidasi et al. [3]. They trained the model with the pairwise ranking loss and achieved an encouraging result compared to the traditional methods. In their following paper, Quadrana et al. [4] designed a parallel RNN architecture to model multi-modal feature representations as data augmentation to improve the personalized recommendation accuracy. Another pure RNN-based model is proposed by Wu et al. [5], which employed the LSTM to solve the personalized recommendation problem.

CNN-Based SRS. Unlike the recurrent architecture in RNN, CNN is well known as a locally connected network to act as “feature extractors”. The typical work applying

CNN in the session recommendation task is proposed by Tang and Wang [6]; in their model (named Caser), CNN takes the whole interaction matrix as an image with time and space, then performs horizontal and vertical convolution separately to extract features. To solve the CNN's capable ability for long-term dependencies issues, Yuan et al. [7] introduced the holed convolution by increasing the receptive field to make up the shortcoming. Recently, Yan et al. [8] encoded the sequence embedding as a three-way vector and employ a 2-D convolution to capture the complex long-range dependencies. Yet the issue of long-term dependencies still exists in CNN because continuously increasing the size of the convolutional kernel filter size will make CNN ineffective to capture the short-term dependencies.

Hybrid SRS. Though these SRSs built on a pure model/technique showed their effectiveness, there are still many limitations with these basic models. To this end, some more powerful hybrid models were proposed to address the particular challenges. For example, Li et al. [9] proposed an encoder–decoder model that combines RNN and attention mechanism for the session recommendations. Jannach and Ludewig [10] proposed to enhance the session representation by its k -nearest neighborhood sessions, which is a hybrid model based on the k -nearest neighbors and RNN. Guo et al. [11] proposed a hybrid model based on matrix factorization and RNN for music recommendation. Xu et al. [12] designed a combination of CNN and RNN, in which all item embeddings were first fed into the RNN to generate the hidden state, and then CNN was applied to search the significant local features. This method also suffered from noise issues. Bach et al. [13] decomposed the original sequences into multiple sub-sequence blocks, then applied CNN to generate union features from each block, these local features were regarded as the timestamp input of GRU to compute the probability among all the candidates. This method verified the importance of the local union features in the recommendation task; however, the pooling operation in CNN will miss the item's features information. Recently, several researchers applied graph neural networks (GNNs) [14–16], especially gated graph neural networks (GGNNs) to generate item embeddings [17–19]. Although graph-based methods are helpful in mining high-order relationships, these methods usually require a huge amount of memory to store and update the node features.

Unlike the aforementioned methods, we incorporate RNN and CNN in an innovative way. Our method fed the whole sequence into CNN and generated multiple context-aware subsequences as the dual-channel input separately. Thus, the long-term and short-term dependencies will be effectively captured. We performed extensive experiments to verify the effectiveness of our method.

2 The Proposed Approach

Our proposed D-CRNN incorporates CNN to GRU to learn sequential features. The task of the session-based recommendation is formulated as follows. Let $S = [s_1, s_2, \dots, s_N]$ denotes the set of all anonymous session sequences, let

$I = \{i_1, i_2 \dots, i_M\}$ represents all unique items collected from S . An session s can be represented as $s = \{i_1, i_2 \dots, i_t\}$, where i_t denotes the t timestamp clicked item in s . We fixed the length of the training sequences within L , and repeatedly add vector $\mathbf{0}$ if there are not enough items in a session.

The architecture of D-CRNN is illustrated in Fig. 1. D-CRNN has three main components: (1) convolution layer, (2) GRU layer, and (3) channel selection layer. The convolution layer is employed to transform the original sequence into the GRU input. Then, the GRU layer learns the long-term dependencies from these subsequences. Finally, the target-aware module adaptively selects the channels to generate the final session representation.

Convolutional Layer. Given the session sequence $s = \{i_1, i_2 \dots, i_t\}$, D-CRNN first map them into a continuous, lower-dimensional space by an embedding lookup table. Then we apply n different sizes of filters to explore the local correlation with multi-scale feature interactions separately. Let $F^k \in \mathbb{R}^{h \times d}$, where $1 \leq k \leq n$, n is the total number of channels, and d represents the item embedding dimension, and $h \in \{1, \dots, L\}$ is the size of the filter. If $F^k \in \mathbb{R}^{1 \times d}$, CNN can be regarded as a point-level convolution; when $h > 1$, a significant signal feature will be picked up regardless of location, which can be regarded as a union-level representation of the center item and its contextual feature. In each convolution, the filter F will slide from top to bottom over the embedding layer output, and interact with the items i (where $1 \leq i \leq L - h + 1$) to generate the result as follows

$$c_i^k = f(O_{i:i+h-1} \odot F^k) \tag{1}$$

Here F^k is the convolutional kernel, and $O_{i:i+h-1}$ is the sub-matrix from the embedding layer, the index of O is the size of the interacting field, \odot is the inner product operator, and $f(\cdot)$ denotes the activation function ReLU.

GRU Layer. After the CNN operation, the original embedding matrix has been transformed into a set of fixed length and union-level feature matrices. Now, we

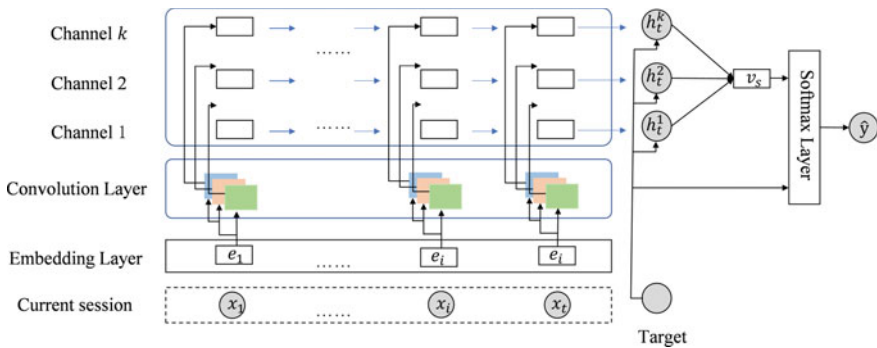


Fig. 1 Framework of D-CRNN model

describe how to feed these union-level features into the RNN layer to capture the long-term dependencies. Specifically, a recurrent network GRU (which is a variant of RNN) will be employed on the CNN output to obtain the hidden state h^k . Here, we choose GRU instead of RNN and LSTM, because the gating mechanism is effective to avoid the exploding/vanishing gradient problem in modeling long-range dependencies and some works showed that GRU performs better than LSTM in the session-based recommendation task [3, 9].

Different from the traditional RNN, where the input x_t demonstrates the item interacted at timestamp t , our input vector $x_t \in \mathbb{R}^d$, $1 \leq t \leq L - h + 1$ denotes the t th convolution operation in the CNN, as described in the convolution layer. We use GRU with the gating mechanism to remember the former long-distance item state information. The detail process of the update state is as follows.

$$z_t = \sigma(W_z c_t^k + V_z h_{t-1}^k + b_z) \quad (2)$$

$$r_t = \sigma(W_r c_t^k + V_r h_{t-1}^k + b_r) \quad (3)$$

$$\tilde{h}_{t-1}^k = \tanh(W_h c_t^k + V_h(r_t * h_{t-1}^k) + b_h) \quad (4)$$

$$h_t^k = (1 - z_t) * h_{t-1}^k + z_t * \tilde{h}_{t-1}^k \quad (5)$$

where $*$ denotes the Hadamard product, σ and \tanh are the activation functions, and W , V , and b are trainable parameters in the current channel.

Channel Selection Layer. Previous work usually aggregates the multi-channel final hidden states h_t^m ($1 \leq m \leq k$) of the k channels with a mean-value attention mechanism, max pooling, or mean pooling. In this paper, we propose to dynamically combine multi-channel embedding to construct final session embeddings by a target-aware attention mechanism. All candidate items are regarded as the targets in this model. Specifically, we use a target attention module to compute weighted value between all channels and each target item $v_i \in I$, its embedding $h_{v_i} \in \mathbb{R}^d$:

$$\beta_{i,m} = \text{softmax}(e_i, m) = \frac{\exp(h_t^m W h_{v_i})}{\sum_{m=1}^k \exp(h_t^m g_{v_i})} \quad (6)$$

where W is the nonlinear parameters. Then, we aggregate all channels with the weighted factor $\{\beta_{i,m}\}$ to build a dual-channel embedding s_h :

$$s_h = \sum_{m=1}^k \beta_{i,m} * h_t^m \quad (7)$$

where $\beta_{i,m}$ is the concentration weight of the target item h_{v_i} . That is, $\beta_{i,m}$ weights the target item to construct the embedding v_C that probably outputs the target item h_{v_i} .

Model Optimization. Given the session embedding s_h , we compute the recommendation probability distribution between all candidate item h_{v_i} and s_h :

$$\hat{y}_i = \frac{\exp(h_{v_i}^T s_h)}{\sum_{j=1}^{|V|} \exp(h_{v_j}^T s_h)} \quad (8)$$

where \hat{y}_i is the recommendation score of all candidates at the next time. At last, we optimize the cross-entropy as the objective function and minimizing the loss to train the parameters:

$$\text{Loss}(\hat{y}) = -\sum_{i=1}^N y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \quad (9)$$

3 Experiments and Analysis

Datasets. We adopted two datasets, namely Yoochoose, Diginetica. In particular, the Yoochoose dataset is collected from the ecommerce website Yoochoose.com. The Diginetica dataset is users' transaction data log.

For Yoochoose, we evaluate the D-CRNN on the most commonly used sub-dataset 1/64, 1/4. Similar to Li et al. [9], sessions from the latest week were used for testing, and others considered as the training data. In addition, given a session $S = [v_1, v_2, \dots, v_n]$, we split all the data sequence to produce the training session and the predict item, i.e., $([v_1], v_2)$, $([v_1, v_2], v_3)$, \dots , $([v_1, v_2, \dots, v_{n-1}], v_n)$. The statistics about the three datasets are shown in Table 1.

Experimental Setup. The number of the batch size is 512; the dimension of embedding size is 50; the hidden size of GRU is 50. The maximum length our model can deal with is 59. We choose three channels with corresponding filter sizes $f = [1, 2, 4]$. We train and optimize our method with Adam. The learning rate $\text{lr} = 0.005$ and decays by 0.1 after every 3 epochs. Dropout layers are used to avoid overfitting: one is after the embedding layer with dropout = 0.3, and the other is after the GRU layer with dropout = 0.5.

Table 1 Statistics of the datasets in our experiments

Datasets	Yoochoose 1/64	Yoochoose 1/4	Diginetica
All clicks	557,248	8,326,407	982,961
Train clicks	369,859	5,917,746	719,470
Test clicks	55,898	55,898	60,858
Num of items	16,766	29,618	43,097
Avg. items	6.16	5.71	5.12

Baselines. We compare and analyze the proposed model with seven representative baselines, including conventional methods and deep learning algorithms:

1. S-POP: S-POP recommends the next-click based on item popularity of the current session.
2. BPR-MF [20]: BPR-MF predicts the next item by decomposing the original user-item interaction matrix and used a pairwise objective function to optimize the matrix factorization.
3. GRU4REC [3]: GRU4REC is the representative work of RNN-based methods in the session-based recommendation task.
4. Caser [6]: Caser is a pure model based on CNN to solve sequential problems.
5. FPMC [21]: FPMC incorporates the Markov chain into matrix factorization to solve the personalized recommendation problem.
6. STAMP [22]: STAMP is a hybrid model based on MLP and attention mechanism.
7. NARM [9]: NARM is a hybrid model based on RNN and attention mechanism.

Evaluation Metrics and Experimental Setup. Two widely metrics are used to evaluate the method performance: Mean Reciprocal Rank (MRR)@10, and Recall@10.

Performance Comparison. Table 2 presents the performance of all methods, where we highlighted in boldface of the best result and underlined the best results of the benchmark to make the comparison clearer. From Table 2, we have the following findings:

1. Although S-POP directly regard the most popular item in the current session as the next one without any complicated statistic or deep learning method, it is not the worst, and is even better than BPR-FM and FPMC, especially on Diginetica. The common problem in FPMC and BPR-FM is that they both ignore the contextual information in the current sequence and cannot deal with the dynamic change of user interests.

Table 2 Performance comparison of D-CRNN with baseline methods

	Yoochoose 1/64		Yoochoose 1/4		Diginetica	
	Recall@10	MRR@10	Recall@10	MRR@10	Recall@10	MRR@10
S-POP	15.31	13.09	14.80	12.79	9.43	7.28
BPR-MF	22.93	10.24	29.30	13.89	4.21	1.89
FPMC	36.12	18.54	37.44	20.05	15.01	6.20
GRU4REC	52.43	24.53	55.49	26.05	17.93	7.73
CASER	59.09	28.08	57.29	28.33	32.64	13.92
STAMP	52.96	25.17	57.67	28.32	33.98	14.26
NARM	57.83	27.42	57.98	28.51	35.44	15.13
Our method	59.79	28.90	60.11	28.94	37.52	15.86

2. Deep learning methods (with the basic model or the hybrid models) consistently performed better than the traditional methods. Compared to the traditional method, deep learning methods are good at capturing complicated correlations. Among these methods, Gre4REC and CASER are the pure models based on RNN and CNN, while others are hybrid models. It demonstrates that a hybrid method is usually a more powerful model when solving solve the same issue. GRU is also a powerful tool for sequential patterns than MLP. It is worth mentioning that although CASER cannot capture the sequential patterns, it also achieves a comparable result. The main reason is that most of the sessions are short due to the limitation of session duration, and items in the sequences are not strictly ordered, so CNN performs better than RNN.
3. D-CRNN achieved the best results in terms of Recall and MRR, outperforming conventional methods and basic neural network models; especially on the Diginetica dataset, D-CRNN improves by 5.8% and 4.8% respectively in terms of Recall@10 and MRR@10. This ascertains the assumption of our method that the operation of CNN can filter the irrelevant features, thus making RNN performs well.

4 Conclusion

In this paper, we designed a hybrid model based on RNN and CNN, which can leverage the advantage of both RNN and CNN and overcome the shortcoming of the single model for the session recommendation. Moreover, the extensive experiment showed the importance of the union-level features and sequential patterns in the sequence prediction task. In the next work, we plan to apply the recent hot graph neural networks to further exploit high-order relationships to improve the accuracy of the recommendation.

References

1. Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Comput* 9:1735–1780
2. Cho K, van Merriënboer B, Bahdanau D, Bengio Y (2014) On the properties of neural machine translation: encoder–decoder approaches. In: *Proceedings of SSST-8, eighth workshop on syntax, semantics and structure in statistical translation*, pp 103–111
3. Hidasi B, Karatzoglou A, Baltrunas L, Tikk D (2016) Session-based recommendations with recurrent neural networks. In: *4th international conference on learning representations (poster)*
4. Quadrana M, Karatzoglou A, Hidasi B, Cremonesi P (2017) Personalizing session-based recommendations with hierarchical recurrent neural networks. In: *Proceedings of the 11th ACM conference on recommender systems*, pp 130–137
5. Wu CY, Ahmed A, Beutel A, Smola AJ, Jing H (2017) Recurrent recommender networks. In: *Proceedings of the 10th ACM international conference on web search and data mining*, pp 495–503

6. Tang J, Wang K (2018) Personalized top-n sequential recommendation via convolutional sequence embedding. In: Proceedings of the 11th ACM international conference on web search and data mining, pp 565–573
7. Yuan F, Karatzoglou A, Arapakis I, Jose JM, He X (2019) A simple convolutional generative network for next item recommendation. In: Proceedings of the 12th ACM international conference on web search and data mining, pp 582–590
8. Yan A, Cheng S, Kang WC, Wan M, McAuley J (2019) CosRec: 2D convolutional neural networks for sequential recommendation. In: Proceedings of the 28th ACM international conference on information and knowledge management, pp 2173–2176
9. Li J, Ren P, Chen Z, Ren Z, Lian T, Ma J (2017) Neural attentive session-based recommendation. In: Proceedings of the 2017 ACM conference on information and knowledge management, pp 1419–1428
10. Jannach D, Ludewig M (2017) When recurrent neural networks meet the neighborhood for session-based recommendation. In: Proceedings of the 11th ACM conference on recommender systems, pp 306–310
11. Guo L, Yin H, Wang Q, Chen T, Zhou A, Quoc Viet Hung N (2019) Streaming session-based recommendation. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining, pp 1569–1577
12. Xu C, Zhao P, Liu Y, Xu J, Sheng VSSS, Cui Z, Zhou X, Xiong H (2019) Recurrent convolutional neural network for sequential recommendation. In: The world wide web conference 2019, pp 3398–3404
13. Bach NX, Long DH, Phuong TM (2020) Recurrent convolutional networks for session-based recommendations. *Neurocomputing* 411:247–258
14. Yu F, Zhu Y, Liu Q, Wu S, Wang L, Tan T (2020) TAGNN: target attentive graph neural networks for session-based recommendation. In: Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval, pp 1921–1924
15. Pan Z, Cai F, Chen W, Chen H, de Rijke M (2020) Star graph neural networks for session-based recommendation. In: Proceedings of the 29th ACM international conference on information & knowledge management, pp 1195–1204
16. Wang Z, Wei W, Cong G, Li X-L, Mao X-L, Qiu M (2020) Global context enhanced graph neural networks for session-based recommendation. In: Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval, pp 169–178
17. Rendle S, Freudenthaler C, Gantner Z, Schmidt-Thieme L (2009) BPR: Bayesian personalized ranking from implicit feedback. In: Proceedings of the 25th conference on uncertainty in artificial intelligence, pp 452–461
18. Rendle S, Freudenthaler C, Schmidt-Thieme L (2010) Factorizing personalized Markov chains for next-basket recommendation. In: Proceedings of the 19th international conference on world wide web, pp 811–820
19. Liu Q, Zeng Y, Mokhosi R, Zhang H (2018) STAMP: short-term attention/memory priority model for session-based recommendation. In: Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining, pp 1831–1839
20. Rendle S, Freudenthaler C, Gantner Z, Schmidt-Thieme L (2009) BPR: Bayesian personalized ranking from implicit feedback. In: Proceedings of the 25th conference on uncertainty in artificial intelligence, pp 452–461
21. Rendle S, Freudenthaler C, Schmidt-Thieme L (2010) Factorizing personalized Markov chains for next-basket recommendation. In: Proceedings of the 19th international conference on world wide web, pp 811–820
22. Liu Q, Zeng Y, Mokhosi R, Zhang H (2018) STAMP: short-term attention/memory priority model for session-based recommendation. In: Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining, pp 1831–1839

Reuse Your Old Smartphone: Automatic Surveillance Camera Application



Lap-Kei Lee, Ringo Pok-Man Leung, and Nga-In Wu

Abstract The life cycle of a smartphone is decreasing rapidly, which leads to a lot of electronic waste and causes damages to the environment. Home security has also become an important concern due to the frequent occurrences of burglaries and home accidents. Surveillance cameras are commonly deployed to improve home security. This paper presents the design of an automatic surveillance camera application called iEye. It aims to reuse components of an old smartphone including the camera, sensors, and microphone to automatically monitor users' home security. iEye is a cross-platform application, which contains a Java application as a server, and an Android app installed on old smartphones as cameras, and installed on the user's smartphone or personal computer as a viewer, respectively. It turns an old smartphone into a home security camera, supporting real-time streaming such that users can see live video of their home on their smartphones anywhere and anytime. iEye also allows users to define abnormal events based on its detection components, namely motion detection, face recognition, human detection, and light detection. Upon an abnormal event, the camera video will be recorded and the user will also be immediately notified by email. iEye is particularly suitable for solitary elderly, working parents, and pet keepers. It also helps to reduce electronic waste.

Keywords Home security · Smartphone · Mobile app · Surveillance camera · E-waste reduction

L.-K. Lee (✉)

School of Science and Technology, Hong Kong Metropolitan University, Ho Man Tin, Hong Kong SAR, China

e-mail: lklee@hkmu.edu.hk

R. P.-M. Leung

The Executive Centre Limited, Central, Hong Kong SAR, China

N.-I. Wu

College of Professional and Continuing Education, Hong Kong Polytechnic University, Kowloon, Hong Kong SAR, China

e-mail: ngain.wu@cpce-polyu.edu.hk

1 Introduction

Smartphone has become a key device for communication nowadays. Yet the life cycle of a smartphone is less than two years in developed countries [1]. Once a smartphone reaches the end of its lifespan, it is usually withheld from the recycling system [2, 3] and becomes electronic waste. Waste electrical and electronic equipment (WEEE) causes a lot of damages to the environment and communities, e.g., they may be illegally exported to countries with unsafe landfilling; their informal recycling may cause pollution and health problems for the communities [4]. A solution to reuse old smartphones can mitigate the problem.

Home security has also become an important concern due to the frequent occurrences of burglaries and home accidents. Surveillance cameras are commonly deployed to improve home security. However, setting up closed-circuit television (CCTV) and hiring a security guard are too expensive and thus not affordable for common families. An alternative solution is to install a home security camera (HSC), which is a home wireless Internet-of-Things (IoT) device with a camera connected to the Internet via WiFi such that the camera owner can watch the live feed through a mobile app provided by the camera provider anywhere. The market of HSCs has been predicted to reach \$1.3 billion by 2023 [5].

Our contribution. Given the needs of HSCs and reusing old smartphones, this paper presents the design of an automatic surveillance camera application called iEye, which re-utilizes the camera, microphone, and sensors of an old smartphone to keep users' homes safe under automatic surveillance. iEye is a cross-platform application containing a Java application as a server, an Android app installed on old smartphones as cameras, and the same Android app on the user's smartphone or personal computer in use as a viewer, respectively. It turns an old smartphone into a home security camera, supporting real-time streaming such that users can see live video of their home on smartphones anywhere and anytime. iEye has various detection components including motion detection, face recognition, human detection, and light detection. Users can define abnormal events for the above detectors for iEye to record the video and notify them by email immediately. A preliminary evaluation on 30 users showed that iEye helps increase the life cycle of smartphones by turning them into HSCs.

1.1 Related Work

There are many IoT systems for home security. Many systems consisted of IP cameras and/or microcontrollers like Raspberry Pi and Arduino, and some of them support monitoring and control through a mobile app (see the survey [6] and the references therein, and the works [7–12] for example). Mahler et al. [13] made use of the onboard sensors of an old smartphone to build a home security system that can monitor door-related events. Their system leverages the accelerometer and magnetometer on a

smartphone and machine learning methods to detect door openings, closings, and rotations, and notify the homeowner via email, SMS, and phone calls upon break-in detection. Jain et al. [14] developed an intelligent automated real-time surveillance that combines a computationally cheap object tracking algorithm and a computationally heavy facial recognition model to meet the real-time processing constraints of a surveillance system.

There are also mobile apps that can turn a smartphone into a security camera. For example, Alfred [15] is a mobile app that provides a live video stream, two-way audio, zoom in, and also supports alerts triggered by motion detection and human detection. Yet it is expensive with a one-time cost of US \$16.99 to remove advertisements and a subscription fee of US \$3.99 per month or \$29.99 per year.

2 The Automatic Surveillance Camera Application—iEye

This section presents the system design and functionality of the automatic surveillance camera application called iEye.

System design. iEye can turn old smartphones running outdated versions of Android (Android 2.3 or above) into security cameras. iEye is cross-platform in the sense that the viewer can be an Android smartphone or a personal computer, which the user is currently using. Figure 1 shows the overall system design of iEye.

Our system requires a server, which is installed with an Nginx server and an Apache Tomcat server; both are free, open-source, and provide high performance and stability. The Nginx server is for video streaming; an old smartphone can send the video stream to the Nginx server and then pass it to the viewer app installed in

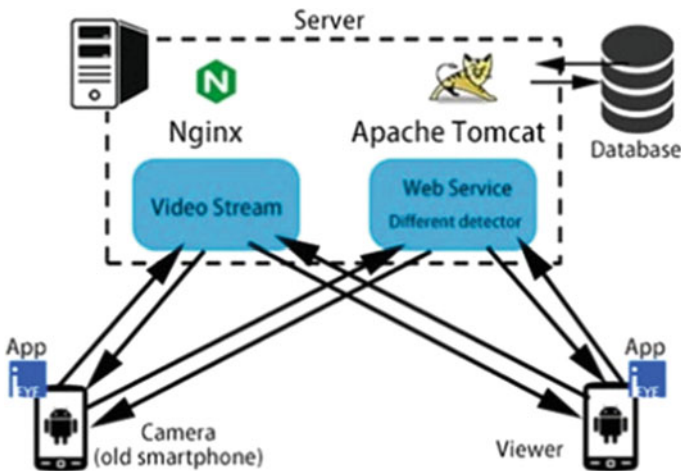


Fig. 1 System design of iEye

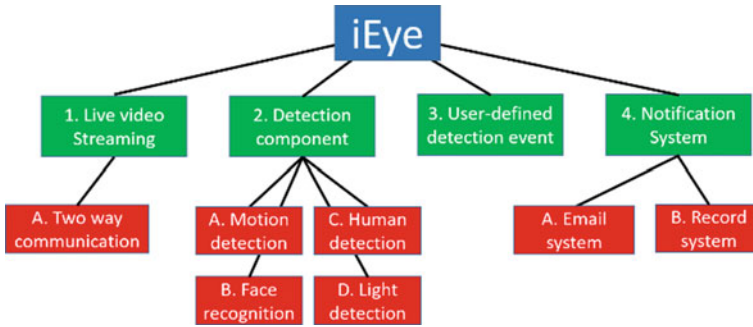


Fig. 2 Functionality of iEye

the user's smartphone currently in use. The Tomcat server is for computation of the detection components; it obtains the video stream from Nginx, executes the image processing and analysis for different detection components, and finally sends the detection result such as coordinates on the video to the viewer app.

Functionality. iEye supports real-time video streaming from the old smartphone camera such that users can see live video of their home on the viewer app anywhere and anytime. It also provides various detection components including motion detection, face recognition, human detection, and light detection. It also supports user-specified detection events that are combinations of events from the four detection components. A notification system component allows event-triggered notifications on the viewer app and email, and the recordings of triggered events. Figure 2 provides a summary of the functionality of iEye.

2.1 Live Video Streaming

The purpose of video streaming is to provide a stable channel to send a compressed video over the Internet and display uncompressed video to a viewer in real time. It plays an important role in iEye since the video quality, stability, and speed directly affect the performance of detection and the video watched by users.

When the user first sets up iEye on a smartphone, the leftmost interface in Fig. 3 is shown for the user to set it as a camera or viewer. iEye supports multiple old smartphones (three at maximum) as the cameras. Real-Time Streaming Protocol (RTMP) is used for establishing and controlling video stream sessions between different endpoints (cameras and viewers). As a prototype, we use QR codes of IP addresses to simplify the connection to the camera and viewer, as shown in the second-left and third-left user interfaces in Fig. 3 (which are self-explanatory). The fourth-left user interface in Fig. 3 shows the viewer interface, where the user can start/stop the live video stream, control the camera and audio communication through the buttons at the bottom (their meanings are shown at the rightmost of Fig. 3). Note that iEye allows

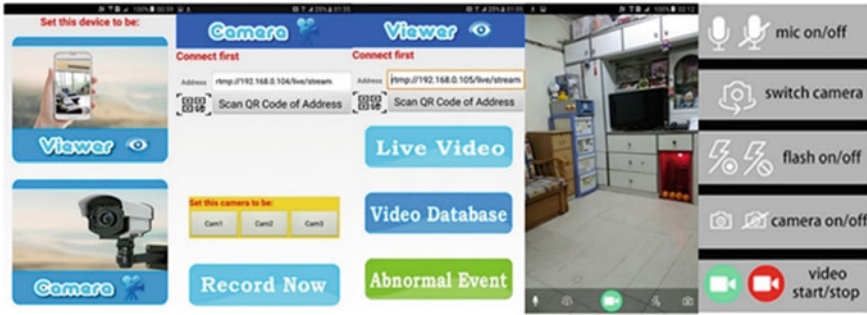


Fig. 3 User interfaces for live video streaming

viewers and cameras to talk to each other (two-way communication); the viewers can click the speaker button to talk to the people at home.

We use Nginx RTMP module and Node Media Client’s SDK and library. Nginx provides RTMP streaming server to send real-time video. Node Media Client provides the video encoder and decoder.

As shown in Fig. 4, the video, captured from an old smartphone, is encoded into H.264 compression for network transmission and sent to the Nginx server. Then, the viewer gets the compressed video data from the server and decodes the data for viewing.

iEye supports multiple viewers and three cameras at maximum. Figures 5 and 6 show the viewer’s user interface on smartphones and personal computers, respectively. As a personal computer has a larger screen size, iEye provides more useful information such as current time, location of the device, and the number of cameras and viewers that are currently online (Fig. 6).



Fig. 4 Live video streaming

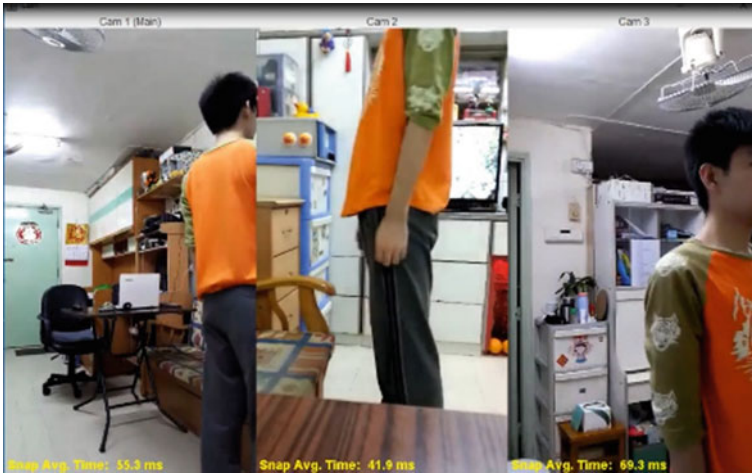


Fig. 5 User interface for multiple cameras

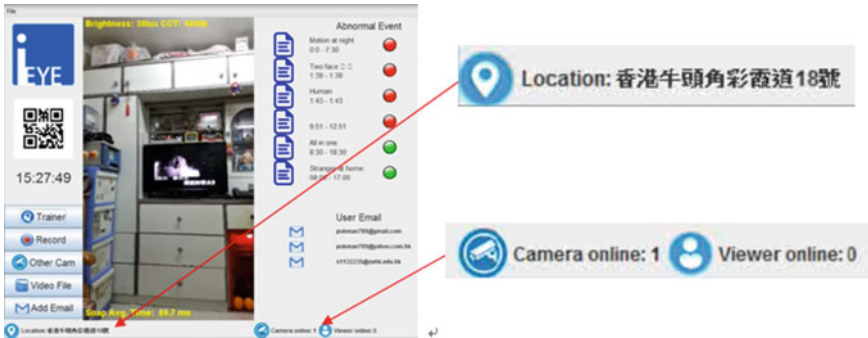


Fig. 6 User interface for the viewer on personal computers

2.2 Detection Components

Motion detection. The motion detection component detects change or movement in a scene. As shown in Fig. 7, a green rectangle is displayed around the motion. The size of the rectangle depends on the area of motion, i.e., a larger motion leads to a larger rectangle.

Motion detection can be done by image differencing of consecutive video frames, foreground or background segmentation, and optical flow. iEye applies the optical flow approach because of its accuracy and efficiency. Optical flow tracks the vector in the pixel level by comparing consecutive frames in a video stream [16]. In dense optical flow, the tracking tries to follow every pixel. It is difficult when the picture lacks detail (e.g., polar bear walking on ice). In sparse optical flow, the tracking only focuses on some pixels that are easy to identify due to their difference in color,

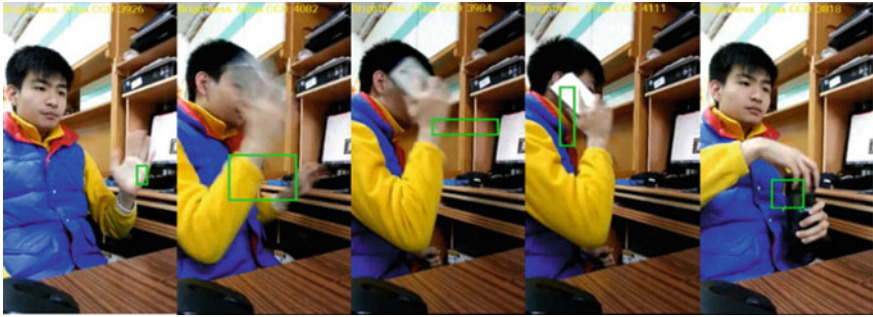


Fig. 7 Motion detection component

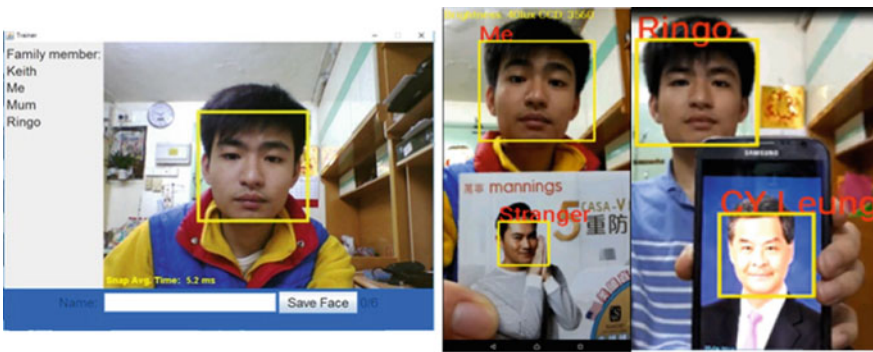


Fig. 8 Face recognition component: face capturing (left); recognition result (right)

texturing, intensity, or color from the pixels nearby. The motion detection component is developed using JavaCV, which supports a number of techniques for the optical flow approach. iEye allows users to set the sensitivity level of the motion from 0 to 10, where a higher sensitivity level can detect smaller motion on the video.

Face recognition. The face recognition component detects and tracks multiple faces as they moved in front of a camera. The user needs to capture the faces of family members to be recognized in the viewer as shown in the left interface of Fig. 8, where six face images in different directions are needed for better accuracy.

Face detection is carried out by a Haar Classifier, pre-trained to find facial features, which is part of JavaCV. As shown in the right of Fig. 8, recognized human faces will be surrounded by a yellow rectangle with a name label; if the face is not recognized to be an existing family member, the name label is “Stranger”.

Human detection. The human detection component detects and tracks multiple humans as they moved in front of a camera. The recognized full body or upper body of a human is surrounded by a yellow rectangle (Fig. 9). Similar to face recognition, human detection is carried out by a Haar Classifier, pre-trained to find body features in JavaCV.



Fig. 9 Human detection component: full body (left) and upper body (right)

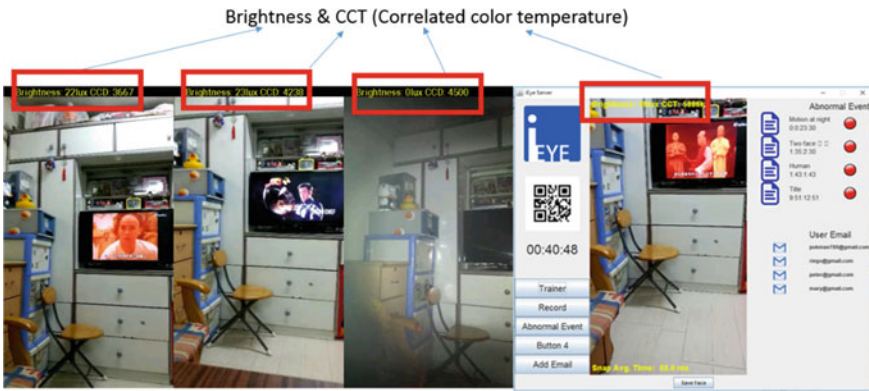


Fig. 10 Light detection component

Light detection. The light detection component provides brightness detail of the user’s home. iEye obtains the brightness value and the Correlated Color Temperature (CCT) index from the light sensor, which can be found nearby almost any smartphone’s camera (Fig. 10). Based on these values, when a dim environment is detected, the flashlight of the smartphone will be turned on automatically so as to ensure the surveillance work in any situation. We can also determine whether the light is on or off at the user’s home.

2.3 User-Defined Detection Event

iEye provides a platform for users to define their own detection event (a.k.a. abnormal event) using the previous detection components. Figure 11 shows the user interface

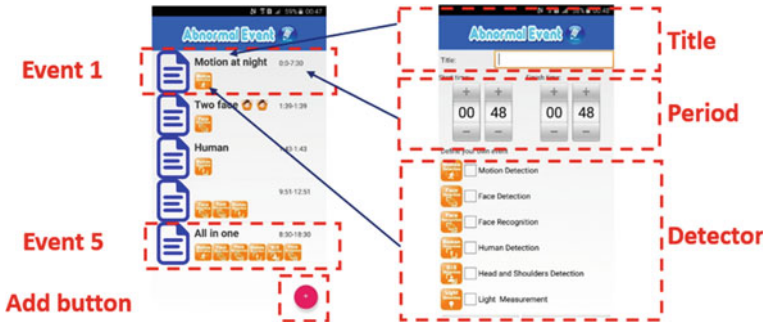


Fig. 11 User-defined detection event

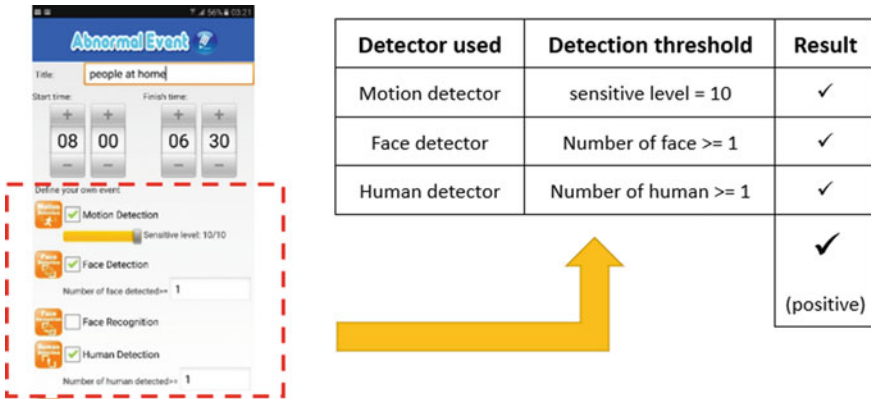


Fig. 12 Creating a user-defined detection event

for user-defined detection events. For example, the event “Motion at night” indicated that it only involves motion detection and the detection period is from 00:00 to 07:30.

To create a new user-defined detection event, the user can click the add button, and then set the event title, different detection components and the corresponding detection period in the user interface shown in Fig. 12 (left). Note that at least one detection must be selected. Let’s take a user who lives alone as an example. If there are no people at home during the working hours from 8:00 a.m. to 6:30 p.m., the user may use three detections such as motion detection, face detection, and human detection to create a detection event called “people at home”. Figure 12 (right) explains the corresponding settings of the detection components. If a detection event is triggered (i.e., we obtain a positive result for the event), it is sent to the server for processing and notifications.

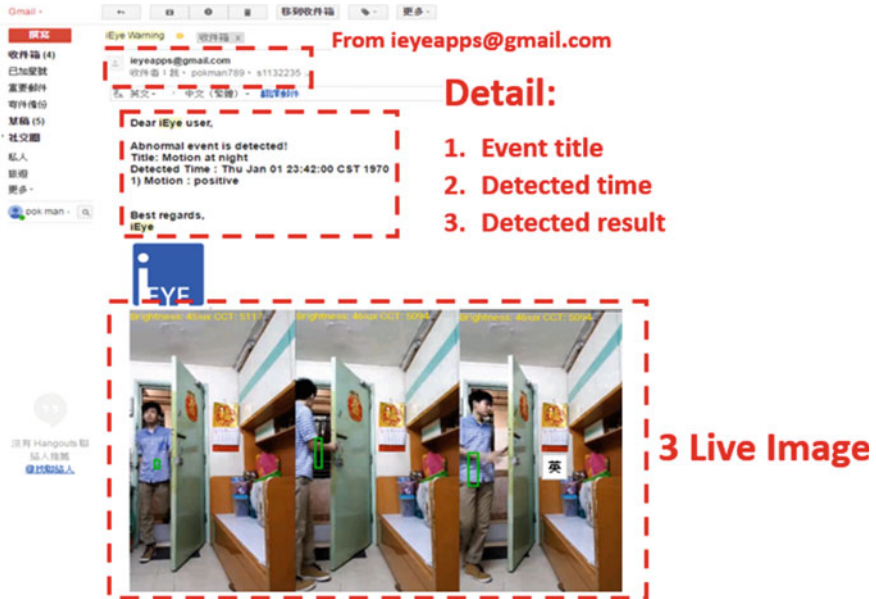


Fig. 13 Notification email

2.4 Notification System

Besides notifying the viewer on smartphones or personal computers, iEye supports notification to user-defined email addresses. Figure 13 shows a notification email.

iEye also contains a record system to record and store digital video in its database. When an abnormal event is triggered, the camera video is recorded and stored in a compressed format, which can be accessed by the viewer application anytime. We also provide a function to set up the maximum storage size for the video (1–32 GB) such that the oldest video will be automatically replaced and deleted.

3 Preliminary Evaluation

We performed a simple user evaluation by inviting 30 participants to use iEye for 15 min and then answer the three questions on a 5-point Likert scale in Table 1.

The majority agreed that iEye can help increase the life cycle of a smartphone (94%), work well as HSCs (100%), and has satisfying functionality (100%).

Table 1 Preliminary evaluation result

Question	Strongly disagree (%)	Disagree (%)	Neutral (%)	Agree (%)	Strongly agree (%)
(1) iEye can increase the life cycle of a smartphone	0	0	6	56	38
(2) iEye can solve your home security problem	0	0	0	37	63
(2) You are satisfied with the functionality of iEye	0	0	0	27	73

4 Conclusion and Future Work

This paper presents the design of an automatic surveillance camera application called iEye, which is a cross-platform application and turns an old smartphone into a home security camera. It supports real-time streaming such that users can see live video of their home on smartphones anywhere and anytime. It also allows users to define abnormal events based on its detection components, namely motion detection, face recognition, human detection, and light detection. Upon an abnormal event, the camera video will be recorded and the user will also be immediately notified by email. iEye is particularly suitable for solitary elderly, working parents, and pet keepers. It also helps to reduce electronic waste. Possible future work directions include enhancing the connection security between the server and clients, e.g., using the algorithm of the secure distributed video surveillance system in [17], and developing new detection components such as fall detection and fire detection to ensure the safety of the elderly and prevent fire accidents.

References

1. Cobbing M, Dowdall T (2014) Green gadgets: designing the future—the path to greener electronics. Greenpeace International, Amsterdam
2. Wilson GT, Smalley G, Suckling JR, Lilley D, Lee J, Mawle R (2017) The hibernating mobile phone: dead storage as a barrier to efficient electronic waste recovery. *Waste Manage* 60:521–533
3. Welfens MJ, Nordmann J, Seibt A (2016) Drivers and barriers to return and recycling of mobile phones. Case studies of communication and collection campaigns. *J Clean Prod* 132:108–121
4. Ongondo FO, Williams ID, Cherrett TJ (2011) How are WEEE doing? A global review of the management of electrical and electronic wastes. *Waste Manage* 31:714–730

5. Market Research Future (2018) Home security camera market research report-global forecast 2023. <https://www.marketresearchfuture.com/reports/home-security-camera-market-3787>
6. Fatima S, Aslam NA, Tariq I, Ali N (2020) Home security and automation based on Internet of Things: a comprehensive review. *IOP Conf Ser Mater Sci Eng* 899(1):012011. IOP Publishing
7. Jain A, Basantwani S, Kazi O, Bang Y (2017) Smart surveillance monitoring system. In: 2017 international conference on data management, analytics and innovation (ICDMAI). IEEE, pp 269–273
8. Aydin I, Othman NA (2017) A new IoT combined face detection of people by using computer vision for security application. In: 2017 international artificial intelligence and data processing symposium (IDAP). IEEE, pp 1–6
9. Tanwar S, Patel P, Patel K, Tyagi S, Kumar N, Obaidat MS (2017) An advanced internet of thing based security alert system for smart home. In: 2017 international conference on computer, information and telecommunication systems (CITS). IEEE, pp 25–29
10. Hutabarat DP, Budijono S, Saleh R (2018) Development of home security system using ESP8266 and android smartphone as the monitoring tool. *IOP Conf Ser Earth Environ Sci* 195(1):012065. IOP Publishing
11. Pawar S, Kithani V, Ahuja S, Sahu S (2018) Smart home security using IoT and face recognition. In: 4th international conference on computing communication control and automation (ICCUBEA). IEEE, pp 1–6
12. Adriano DB, Budi WAC (2018) IoT-based integrated home security and monitoring system. *J Phys Conf Ser* 1140(1):012006. IOP Publishing
13. Mahler MA, Li Q, Li A (2017) SecureHouse: a home security system based on smartphone sensors. In: 2017 IEEE international conference on pervasive computing and communications (PerCom). IEEE, pp 11–20
14. Jain V, Pillai MS, Chandra L, Kumar R, Khari M, Jain A (2020) CamAspect: an intelligent automated real-time surveillance system with smartphone indexing. *IEEE Sens Lett* 4:1–4
15. Alfred [Mobile app] (2021) Retrieved from <https://alfred.camera/>
16. Davison A (2013) Vision-based user interface programming in Java. Amazon Digital Services, Inc
17. Albano P, Bruno A, Carpentieri B, Castiglione A, Castiglione A, Palmieri F et al (2012) A secure distributed video surveillance system based on portable devices. In: International conference on availability, reliability, and security. Springer, Berlin, Heidelberg, pp 403–415

A Model of UAV-Based Waste Monitoring System for Urban Areas



Dalibor Dobrilovic , Gordana Jotanovic , Aleksandar Stjepanovic ,
Goran Jausevac , and Dragan Perakovic 

Abstract This paper presents an approach in using unmanned aerial vehicles (UAVs) with remote imaging for urban waste monitoring. The system is designed to monitor green areas, public trash cans, and unregulated landfills and to detect possible violations of garbage disposal rules. Public green urban areas, such as parks, green surfaces, sport terrains, and bathing areas, are gathering places for people and therefore prone to unregulated waste disposal. The proposed solution describes the real-time monitoring of the area using drones and the detection of irregularities in a garbage disposal. The cameras mounted on drones are used to take images of public targeted areas at pre-mapped points. Visual data collected by supervisor drones are used for further processing and notification of authorized personnel and institutions.

Keywords Internet of Things (IoT) · Urban waste monitoring · UAVs · Supervisor drones · Docking station

D. Dobrilovic (✉)

Technical Faculty “Mihajlo Pupin” Zrenjanin, University of Novi Sad, Zrenjanin, Serbia
e-mail: dalibor.dobrilovic@uns.ac.rs

G. Jotanovic · A. Stjepanovic · G. Jausevac
Faculty of Transport and Traffic Engineering, University of East Sarajevo, Doboј,
Bosnia and Herzegovina
e-mail: gordana.jotanovic@sf.ues.rs.ba

A. Stjepanovic
e-mail: aleksandar.stjepanovic@sf.ues.rs.ba

G. Jausevac
e-mail: goran.jausevac@sf.ues.rs.ba

D. Perakovic
Faculty of Transport and Traffic Sciences, University of Zagreb, Zagreb, Croatia
e-mail: dperakovic@fpz.unizg.hr

1 Introduction

The utilization of new technologies such as the Internet of Things (IoT), UAVs, robotics, and artificial intelligence can play an important role in the efforts to make cities greener, safer, and more efficient. Improving safety and quality of life can be achieved by connecting devices, vehicles, and infrastructure across the city. According to research conducted by the United Nations, the population in cities by 2050 should grow up to about 66% of the total population [1]. On the other hand, the development of modern information and communication technologies provides huge opportunities for the implementation of artificial neural networks and modern communication technologies such as Narrowband Internet of Things (NB-IoT). NB-IoT is a low power wide area network (LPWAN) radio technology standard developed by 3GPP to enable a wide range of cellular services. NB-IoT focuses on outdoor and indoor coverage, low cost, long battery life, and high connection density. For these reasons, this technology is very suitable for implementation in the system of control and management of waste in urban areas. Building such solutions on open standards-based communication platforms that can be used continuously is a serious challenge. The usage of other LPWAN technologies, such as LoRa and LoRaWAN, in combination with UAVs for smart city environments, is considered in [2].

One of the problems of big cities is the problem of waste management and waste collection. It is important to note that currently almost all applications that use the “Internet of Things” for smart waste management focus on commercial waste and municipal, public waste containers, rather than household waste. Efficient waste collection is a necessary service and the application of smart cities. The use of emerging technologies, such as UAVs, can lead to significant improvements in the waste management process. The best technological solutions can be achieved in smart cities by creating different stakeholders to work together. Integration of institutions, utilities, and service companies from different areas is necessary to create the solutions that provide a new quality of life in urban and rural areas.

The traditional way of managing waste collection from waste containers has been to schedule and send large trucks around the city at regular intervals. When a truck arrives at the location of the container, it would be emptied regardless of whether the container was full or half empty. Some of the containers may be empty for a longer period while, depending on the structure and behavior of the citizens, some containers can be filled faster than usual, leading to excessive waste and garbage-related health hazards. The additional problem can be caused by uncontrolled garbage disposal in green areas such as parks, sports grounds, and picnic places. Therefore, these places should be monitored together with the public waste containers. Another big problem is the illegal disposal of waste in places that are not planned for that purpose, resulting in unregulated landfills. Therefore, there is a need for constant monitoring of certain critical points to obtain timely information, providing vital data for further activities. To address all enlisted problems, this paper is presented the solution for the system based on UAVs for monitoring, surveillance, and collection of data from the critical areas.

This paper proposes a UAV-based IoT system that solves the problem of observing and monitoring predefined drone trajectories in urban environments where there are static and dynamic obstacles. The system includes the model with the proposed algorithm of random UAV path planning. The algorithm is based on traveling salesman problem (TSP). We proceed as follows. Section 2 provides an overview of related works. The proposed model of the system for monitoring and identifying garbage in public areas is present in Sect. 3. Section 4 presents the random path planning algorithm of the proposed model in a specified environment. Finally, we present our conclusions and a reference to future research in Sect. 5.

2 Related Work

There is a lot of related research works dealing with the topic of monitoring and controlling urban waste using AI and emerging technologies, as well as dealing with the implementation of UAVs in smart cities applications. In Ref. [3], the authors follow a case study for a city in the Republic of Iran and use an artificial neural network to predict the generated waste weekly. The authors trained the proposed neural network model using data collected in the period from 2004 to 2007. As a result, it was found that a neural network with 16 neurons in three hidden layers gives the best prediction results.

The paper [4] discusses monitoring the garbage capacity with mobile phones to prevent the overflow of the garbage from the container. The system consists of three garbage robots, namely (G-Bot 1, G-Bot 2, and G-Bot 3). Each of these robots sends the data to the mobile phones, and the collected data can be checked with mobile phones using Blynk application. The goal of the research presented in [5] is to evaluate the effectiveness of UAVs in monitoring landfill settlement in a real post-closure scenario, by comparing two models obtained through the acquisition of UAV imagery from two separate flights, repeated after about 6 months.

The authors of [6] attempt to show how collaborative drones and IoT improve the smartness of smart cities based on data collection, privacy and security, public safety, energy consumption, and quality of life in smart cities. Article [7], presents a real-time and power-efficient air quality monitoring system based on aerial and ground sensing. The architecture of this system consists of the sensing layer to collect data, the transmission layer to enable bidirectional communications, the processing layer to analyze and process the data, and the presentation layer to provide a graphic interface for users. For data processing, spatial fitting and short-term prediction are performed to eliminate the influences of incomplete measurement and the latency of data uploading. This implementation has been deployed in Peking University and Xidian University since February 2018 and has collected almost 100,000 effective values so far.

IoT and UAV technology cooperation play a vital role in green IoT by transmitting collected data to achieve a sustainable, reliable, and eco-friendly Industry 4.0. The survey presented in [8] gives an overview of the techniques and strategies proposed

to achieve green IoT using UAVs infrastructure for a reliable and sustainable smart world. In [9], the authors investigate the possibility of using drones to monitor garbage disposal at unauthorized locations. The proposed system uses machine learning and artificial intelligence techniques to detect illicit waste in images collected by drones. The authors suggest the further research be based on developing an automated robotic system to handle the detected waste. Paper [10] shows the smart system for the detection of garbage with image processing and the usage of a drone to capture images of locations with garbage. The authors use drones, with an Arduino microcontroller device, a deep neural network, the Python programming language, and Google's cloud platform.

Youme et al. [11] presented an automatic solution for the detection of clandestine waste dumps using unmanned aerial vehicle (UAV) images in the Saint Louis area of Senegal, West Africa. This task has a very high spatial resolution of UAV images (on the order of a few centimeters) and an extremely high level of detail, which require suitable automatic analysis methods. The proposed method begins with segmenting the image into four regions, then reducing the size of input images into $300 \times 300 \times 3$ for the CNN entries, and labeling images by determining the region of interest. The results show that the model recognizes targeted areas well but has difficulties with some areas lacking clear ground truths.

The paper [12] discusses the applications of UAVs in smart cities, their opportunities, and their challenges. Considering that UAVs have a wide range of applications in many fields like environmental hazards monitoring, traffic management, and pollution monitoring, all of which contribute greatly to the development of any smart city, authors discussed the challenges and issues such as safety, privacy, and ethical uses of UAVs.

An additional role model for the system proposed in this paper and its further development can be taken from the experience in building automated video surveillance (AVS) systems that are designed to automatically monitor objects in real time. One of these works presents the experience in building the distributed video surveillance system based on a client-server architecture as it is presented in [13]. The proposed system is accessible from portable devices such as tablets and smartphones. Similarly, the paper [14] targets the demand for a realistic wireless AVS system simulation framework that models and simulates most of the details in a typical wireless AVS framework. The proposed simulation framework is built over the well-known NS-3 network simulator.

3 The Model of the System

The model of UAVs-based urban waste monitoring system is presented in this section. The model is designed to monitor green urban areas, using the unmanned aerial vehicle (UAV). The role of UAV is to monitor the city region to prevent illegal waste dumps and containers overloads in public areas. Also, the goal of the model is to create a balanced low-cost and effective solution that will reduce the cost of the deployment

and maintenance, without reducing the efficiency of monitoring. The idea is to use a single or minimized number of UAVs to cover all specified locations. The model is built on the algorithm for UAV path planning based on the traveling salesman problem (TSP) and genetic algorithm (GA). The model assumes the usage of the centralized system. Data processing is placed in the control center, but it is out of the focus of this model. The model is based on the usage of available and commercially popular drones. Furthermore, the equipment of commercial drones plays a significant role in shaping their price. The system has centralized data processing and drone management. The system architecture is based on [14] and consists of preferably one, but possibly more drones (depending on the size of the area to cover), and the control center to detect and identify improperly disposed waste. The system is automated and operates with the help of a drone called the supervisor. The drone supervisor enables the detection of illegal garbage and unregulated garbage dumps on green urban areas. The image processing component is deployed in the control center, but it is not considered in this proposed model.

3.1 Supervisor Drone

The goal of surveillance drones is to observe public areas. The takeoff of the supervisor drones is performed according to a predetermined schedule. Drone monitors locations along defined paths, at an altitude of 30–70 m above the ground. Depending on the size of the terrain and the needs of the system, several surveillance drones can be deployed. Higher flight altitude (e.g., 70 m) is required to reduce the number of static obstacles in process of designing the drone trajectories. Furthermore, the higher altitudes of the drone can make drones hardly visible to the persons who are violating the regulations and can avoid disturbance to the citizens. Drone captures the current state of the monitored area at defined locations. The defined locations will be discussed later (Sect. 4), together with the algorithm used for drone path planning. After capturing images, the drone returns to the control center to upload images for further processing. The monitoring drones should have the following configuration: GPS module, optional sensors to avoid obstacles, Wi-Fi, LTE (4G), or both communication modules (depending on the network infrastructure and communication requirements), and a camera for capturing images. In addition, drones should have support for route tracking. Using the location definition and path to location data, the drone goes to one or more inspection sites, where it captures images and optionally a video. The drone arrives at the position by a dynamically defined path formed in the control center. When the supervisor drone reaches the position, it descends to a height of 3–5 m above the ground to perform a more detailed survey of the location. Captured location images are stored on SD card. After completing the capturing at all defined locations, it returns to the drone docking station, where it loads the data, reloads the logs, and charges the batteries. Data collected with supervisor drone are used to initiate further actions. In addition to GPS, communication modules, and sensors, this type of drone should have support for route programming, a high-resolution camera for capturing images and videos, and larger SD storage for captured images.

3.2 Control Center and the Communication System

The control server receives data from the drone, stores the data in a database, performs analyses, and performs further processing. It should have high data processing capabilities for image processing and processing algorithms for defining drone routes. If the processing module detects an irregular situation, it proceeds with the actions in alarming the authorized personnel. The image processing module and the further actions are not considered in this paper and the presented model. The communication infrastructure of the model is not also considered in more detail. It should provide the connectivity of the PC (Docking Station) with the drone. This connectivity is important for two major tasks. One task is uploading the calculated path to the drone from the PC. The second task is downloading captured images to the control center. A drone Docking Station is a fully automated system that serves drones. A Docking Station is a specific place for the safe vertical takeoff and landing of drones. In addition to this function, it should enable fast charging of drone batteries, establish communication between the control center and the drone, and maintain their airworthiness. It would be desirable for the proposed system to use static stations with a wired connection to the control center and the power supply infrastructure [14]. The important component of the presented model is the module for route calculation located in control center. The process is shown in Fig. 1.

Figure 1 describes the process of route planning and UAV operations. This procedure is designed to enable the use of one or a minimal number of drones for covering the specified areas, thus reducing the UAV fleet costs and maintenance. First, the traveling salesman algorithm (TSP) with genetic algorithm (GA) is used to calculate the path for the UAV route. The calculated route is randomly formed from the set of nodes of interest. The definition of the set of nodes of interest is described in the following section. After the route calculation, the system checks if the route is longer than deployed UAV capability. If the route is longer, it is calculated again. If the route is within the UAV operational range, it is uploaded to the UAV. After uploading the route, the UAV moves according to the schedule, following the uploaded path. UAV takes images at defined locations, and after the ending of the tour, it returns to the control center. The images are uploaded to the system, and the recharge of the batteries is initiated. The presented set of actions is repeated at the defined period.

4 The Path Definition

This section describes the process of path calculation and the node set definition. The first step of the process is node set definition, and it is performed once at the start of system operation, after some upgrade of the system, or change of the waste management policy. The node set depends on the targeted items of the system. The system presented in this paper is designed to monitor the unregulated waste disposal at the places of interest by taking photos at the given locations. The collected photos

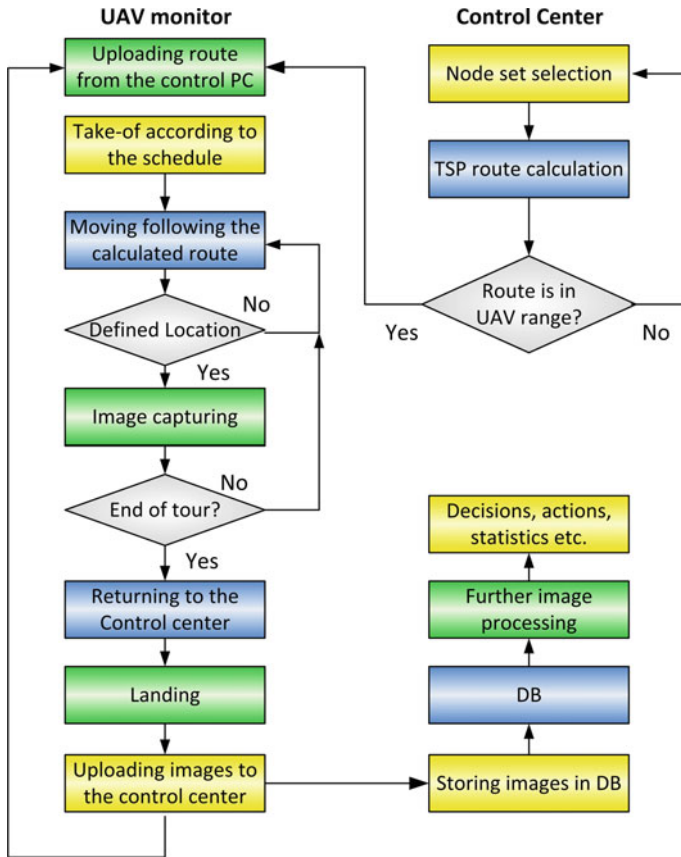


Fig. 1 Route planning and UAV operations algorithm

are further processed in the search of pollution and unregulated waste detection. As places of interest, the gathering places of the people such as parks, green surfaces, sport terrains, and bathing areas are considered. The presented model of the system is designed to randomly select a smaller number of observing locations (10, 15, 20, or 30) to program the drone route each time before sending the drone to the mission. In this way, the locations will be supervised randomly in turns, providing the possibility to monitor a large area with one drone. This will significantly reduce the investment in the system and its maintenance costs.

The example node set is tailored by the city of Zrenjanin. Zrenjanin is a city in Serbia, located in the Central Banat region. Its geographical coordinates are 45° 230N, 20° 2322E, the urban area covers 193.03 km² (74.53 miles²), and the population is 76,511 (2011 census). The observing locations (observing points) have the structure as follows: 29 parks, green surfaces, sport and bathing areas, 97 public garbage container locations, and one unregulated solid waste landfill, see Fig. 2. The

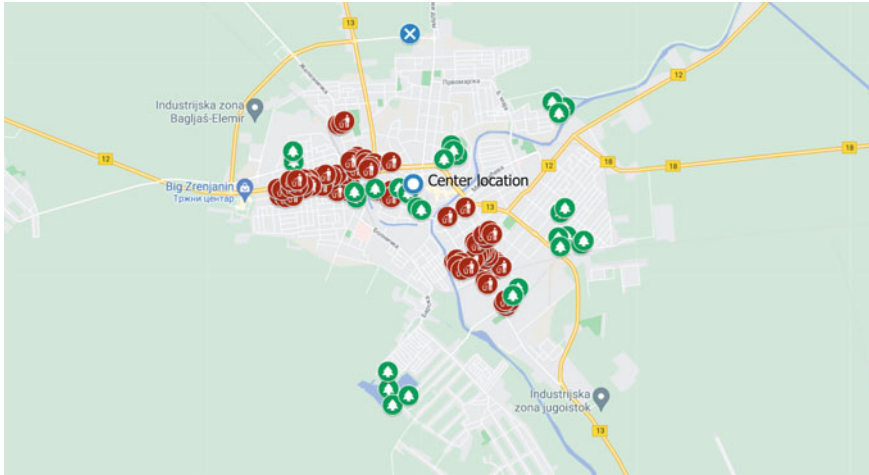


Fig. 2 Observing point locations in the city of Zrenjanin

garbage container places are located near multi-story residential buildings where two or more containers are grouped. Because the citizens dispose of their garbage at these locations frequently, some containers can be over-filled, and the garbage in some cases might be scattered around the container. Only one unregulated solid waste landfill is covered with these locations because such places are located outside the city, and their inclusion in the system coverage area will significantly increase the range of UAV flight, which will require the deployment of UAVs with significantly better performance and thus higher costs.

The central location is used as a headquarters and a placement of control center. It is the starting and ending locations of each route. The number of observing locations is 127 with one central location giving a total of 128 locations. The traveling salesman algorithm (TSP) with genetic algorithm (GA) is used to calculate the path for 10, 15, 20, or 30 randomly selected observing locations [15]. Each calculated route starts at the central location and ends at the same point.

The MATLAB/Octave code is used for the simulation. The code is built upon the [15] and modified according to the presented model. The results of the simulation are presented in Table 1. The simulation uses algorithm with the changeable number of randomly selected nodes, number of iterations (required with the MATLAB/Octave code), and number of tests as well.

The distance for the calculated path for each set of randomly selected nodes is calculated and recorded for the analyses. As one can see from Table 1, the average distances for 10 nodes are 10 km, for 15 and 20 nodes 15–17 km, and for 30 nodes are around 26 km. The results also show that 1000 iterations give accurate results.

In Fig. 3, the two route graphs for 10 and 15 randomly selected nodes are given. The first route is for the following locations: Center-L0011-L0108-L0139-L0135-L0015-L0004-L0185-L0182-L0162-L0106-Center. The second route is as follows: Center-

Table 1 Summary of simulation results

No.	Nodes	Iterations	Tests	Avg. (km)	Min (km)	Max (km)	St. dev. (km)
1	10	1000	100	13.18	12.93	16.08	0.57
2	10	1000	500	13.26	13.14	15.12	0.19
3	15	1000	100	17.27	15.21	22.81	1.84
4	15	1000	500	15.75	14.25	21.72	1.22
5	20	1000	100	16.02	13.83	21.37	1.55
6	20	5000	100	16.38	14.94	21.52	1.38
7	30	1000	100	26.21	19.05	35.10	3.04
8	30	5000	100	26.83	22.42	35.96	2.56

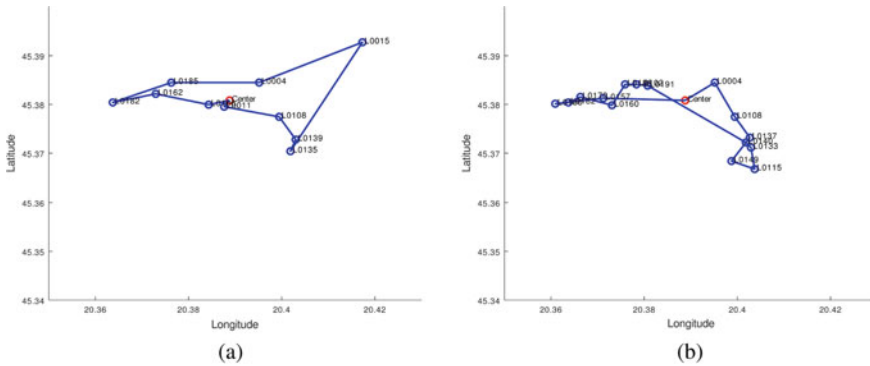


Fig. 3 Calculated routes with **a** 10 nodes and **b** 15 randomly picked nodes

L0004-L0108-L0137-L0149-L0115-L0133-L0140-L0191-L0102-L0188-L0160-L0170-L0182-L0180-L0157-Center. The results show that considering the average route length (up to 15 km) the 10 or 15 locations can be easily monitored with one low-cost UAV.

5 Conclusion

This paper has presented an approach to using UAVs for remote imaging for urban waste monitoring. The presented model of the system is designed to monitor green areas, public trash cans, and unregulated landfills and to detect possible violations of garbage disposal rules. Public green urban areas, such as parks, green surfaces, sport terrains, and bathing areas, are gathering places for people and therefore prone to unregulated waste disposal. Other places of interest for monitoring are locations of public garbage containers located near multi-story residential buildings and possible locations of unregulated waste disposal.

The proposed model includes a method for random path selection allowing utilizing a minimal number of UAVs, as low as only one, for covering the specified area. The idea is to select the random locations for monitoring, thus giving the optimal ratio of drone utilization, areal coverage, and monitoring efficiency. The low number of drones affects the lower system purchase and maintenance costs. The paper has presented a model of system deployment with a general description and a detailed description of the method for path selection and drone operations. The path selection model is based on the traveling salesman problem (TSP) and genetic algorithm (GA).

Further work will include further development of the model. The priority of the model development will be the addition of statistics of locations monitoring frequency, the addition of priorities for certain locations, and modification of the path selection algorithm. The path selection algorithm will be required in cases when drones cannot use direct paths between certain nodes and are forced by the regulation and other factors to follow streets and other restrained paths.

References

1. Pardini K, Rodrigues JJ, Kozlov SA, Kumar N, Furtado V (2019) IoT-based solid waste management solutions: a survey. *J Sens Actuator Netw* 8:5
2. Chen L-Y, Huang H-S, Wu C-J, Tsai Y-T, Chang Y-S (2018) A LoRa-based air quality monitor on unmanned aerial vehicle for smart city. Presented at the 2018 international conference on system science and engineering (ICSSE)
3. Shahabi H, Khezri S, Ahmad BB, Zabihi H (2012) Application of artificial neural network in prediction of municipal solid waste generation (case study: Saqqez City in Kurdistan Province). *World Appl Sci J* 20:336–343
4. Husni NL, Prihatini E, Silvia A (2019) Garbage monitoring and warning system. Presented at the 2019 international conference on electrical engineering and computer science (ICECOS)
5. Baiocchi V, Napoleoni Q, Tesei M, Servodio G, Alicandro M, Costantino D (2019) UAV for monitoring the settlement of a landfill. *Eur J Remote Sens* 52:41–52
6. Alsamhi SH, Ma O, Ansari MS, Almalki FA (2019) Survey on collaborative smart drones and internet of things for improving smartness of smart cities. *IEEE Access* 7:128125–128152
7. Hu Z, Bai Z, Yang Y, Zheng Z, Bian K, Song L (2019) UAV aided aerial-ground IoT for air quality sensing in smart city: architecture, technologies, and implementation. *IEEE Netw* 33:14–22
8. Alsamhi SH, Afghah F, Sahal R, Hawbani A, Al-Qaness A, Lee B, Guizani M (2021) Green internet of things using UAVs in B5G networks: a review of applications and strategies. *Ad Hoc Netw* 102505
9. Anadkat AP, Monisha BV, Puthineedi M, Patnaik AK (2019) Drone based solid waste detection using deep learning & image processing, p 8
10. Achaliya P, Bidgar G, Bhosale H, Dhole P, Gholap K, Chandwad SC (2020) Drone based smart garbage monitoring system using computer vision. *Int J Creat Res Thoughts* 8:6
11. Youme O, Bayet T, Demebele JM, Cambier C (2021) Deep learning and remote sensing: detection of dumping waste using UAV. *Procedia Comput Sci* 185:361–369
12. Mohammed F, Idries A, Mohamed N, Al-Jaroodi J, Jawhar I (2014) UAVs for smart cities: opportunities and challenges. In: 2014 international conference on unmanned aircraft systems (ICUAS)
13. Albano P et al (2012) A secure distributed video surveillance system based on portable devices. In: *Multidisciplinary research and practice for information systems. CD-ARES 2012. Lecture*

- notes in computer science, vol 7465. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-32498-7_30
14. Alsmirat MA, Jararweh Y, Obaidat I, Gupta BB (2017) Automated wireless video surveillance: an evaluation framework. *J Real-Time Image Proc* 13:527–546. <https://doi.org/10.1007/s11554-016-0631-x>
 15. Jausevac G, Dobrilovic D, Brtko V, Jotanovic G, Perakovic D, Stojanov Z (2021) Smart UAV monitoring system for parking supervision. In: *Future access enablers for ubiquitous and intelligent infrastructures*. Springer International Publishing, Cham, pp 240–253
 16. Chernov V (2013) Solution of traveling salesman problem (TSP) using genetic algorithm (GA)

A Secure Multicontroller SDN Blockchain Model for IoT Infrastructure



K. Janani and S. Ramamoorthy

Abstract IoT is making significant progress in a variety of fields, including health care, smart grids, supply chain, and so on. It also makes people's daily lives easier and improves their interactions with one another and their surroundings and environment. There is a variety of research on decentralized computing for IoT develops a decentralized IoT-based biometric facial recognition solution for COVID-19 lockdown cities. They propose a three-layer architecture (application layer, control layer, and data layer) and then create a blockchain framework on top of it to entirely restrict public movements. The software-defined network is the most widely utilized solution for establishing secure network interaction and building secure IoT infrastructures. They give a solid and dependable framework for dealing with dangers and issues like security, scalability, and confidentiality. This study provides a blockchain-based software-defined IoT framework for smart networks that are optimized for energy efficiency and security. Indeed, multicontroller SDN blockchain (MC-SDNBC) has been extensively used to manage vast-scale networks which are, though, subject to a variety of attacks, include false data injection, which causes regulator topology inconsistencies. Every software definition network domain is administered with a single master controller who communicates with both the masters of the other Internet via blockchain. The controller unit generates blocks of dynamic network modifications, which are subsequently evaluated by redundant controllers using a reputation technique given by the control system. The popularity system uses continuous and coupled reactive fading reputé algorithms to score the controllers, for example, the voter's maker and block, during each voting activity. The analysis findings show that false flow rule insertion may be detected quickly and efficiently, keeping more secured IoT Systems.

Keywords IoT · MC-SDN · Blockchain · Security

K. Janani (✉) · S. Ramamoorthy
Department of Computer Science and Engineering, SRM Institute of Science and Technology,
Kattankulathur, India
e-mail: jk6005@srmist.edu.in

S. Ramamoorthy
e-mail: ramamoos@srmist.edu.in

1 Introduction

According to a survey titled “State of IoT Security,” attacks on the Internet of Things surged by 22% in the last quarter. According to the survey, some sectors, such as smart infrastructure, smart cities, healthcare, banking, and transportation, have the highest assault risk. Attacks are more complex and elevated by the day, which would be a cause for alarm. Blockchain, which has six main features decentralized, irreversible, transparent, autonomous, anonymity, and free software [1], has emerged as one of the modern approaches acknowledged by both research and industry in the last decade. Likewise, the Internet of Things (IoT) is a promising technology field in which many smart applications are being developed. IoT devices are implemented using actuators, intelligent devices, and sensors. The physical layer, network layer, and application layer are the three layers that make up the IoT system’s core architecture [2].

Considering the worldwide health catastrophe COVID-19, businesses are eager to grow up work-from-home possibilities with heavy security and all focus specifically. As a result, remote management usage is more important than ever. Different heterogeneous devices are connected and communicated with each other in an IoT application [2]. Because the number of connected things to the Internet is increasing these days, managing and controlling IoT has become a difficult task. SDN steps in to provide the IoT network’s adaptability and scalability without requiring existing implementations to change their design [3]. Because the majority of smart gadgets are low end, they are more vulnerable to attacks. There is a requirement for lightweight algorithm for cryptographic provision of a safe, and computing to create IoT-based communication services. The confidentiality, integrity, and availability (CIA) primary security purpose must be kept updated by the application. With the growing popularity of blockchain technology, increasing study has focused on the use of blockchains in conjunction with SDN, allowing untrustworthy persons to connect with others in a suitable area without the need for a trusted third party [4].

Blockchain is another sophisticated technology that can be combined with SDN-based IoT applications. Blockchain is a developing decentralized technology that can be integrated with SDN-based IoT systems. Every block of the process is continuously saved, and several blocks are chained together through controlling hash values. Using this blockchain technology will boost security and privacy. Several academics have made numerous recommendations for improving the performance of the network, but none of them can resolve the issue. Even though the Internet of Things, software-defined networks, and blockchain technologies are being merged to provide a better solution for smart infrastructure devices, those technologies also can enable dependable data transfer and interaction in networks. However, when these technologies are used, they add to the complexity. Many authors have explored many different solvents. A few of these technologies give a significant level of protection, but they are not a feasible approach [5].

A distributed blockchain-based SDN-IoT-enabled infrastructure for smart buildings is proposed in this paper. In this regard, smart buildings serve as a dependable domain for automatically controlling and managing temperature, security, light-

ing, and other building functions. Furthermore, SDN-based smart buildings include important factors such as goals, technique scope, target design (centralized network controller), networking devices, and resources configuration (homogeneous and heterogeneous). Security, energy efficiency, network monitoring, reliability, QoS, and delay reduction are some of the main goals. WiFi, LiFi, Zigbee, and Bluetooth are the communication technologies for SDN-enabled smart buildings [6].

To address the IoT security dilemma, we provide an infrastructure security that blends blockchain using a multicontroller software definition network. The key notion of the design and architecture is to allocate a set of controllers from each domain [7], which employs a large number of control systems to provide error detection, and our design focuses on ensuring safe and reliable inter-controller interactions. To achieve this purpose, the system design incorporates a controller unit and numerous controllers for each network domain. Each controller could be the owner in one domain, but it may be duplicated in another. The duplicate controllers select whether or not to validate the nodes of network architecture improvements generated by the control board. The design also includes a reputation system that uses constant and dynamic fading reputation algorithms to rate the controllers after every voting activity. Malicious master controls and duplicate controllers that offer false voting would be identified in this method. The following are the paper's primary achievements in further detail [8].

1. To secure inter-controller interaction, we present multicontroller block (MCB-SDN), IoT privacy issues design which combines software definition network and blockchain technology. Every domain is provided a special master device and several redundant controllers via MCB-SDN. The control systems are blockchain users; the master controller generates blocks, and the redundant directors monitor its activity (Fig. 1).
2. In MCB-SDN, we include a credibility process that rates controllers using one of two methods: (1) constant fading credibility, which allows the control system to forget past operational activities at a steady speed, or (2) simultaneous adaptive fading credibility, which ranks the console which uses various constants based also on device's credibility, gets in trouble, the faster good experiences fade away. On either side, the better the control behaves, the much more quickly unfavorable experiences fade away.
3. Analysis methods, including Mininet software products, ONOS, and multicontroller SDN-Chain, are used to execute the suggested MCB-SDN design. MCB-SDN archive low detection delay and it allows user to identify all maliciously inserted attacks. According to the findings the proposed MCB-SDN model provides dynamic nature of threat detection time to identify rogue in the network dynamic nature of the detection time to identify. Furthermore, the reputation approach provides for flexible detection time of rogue devices based on the network executive's needs.

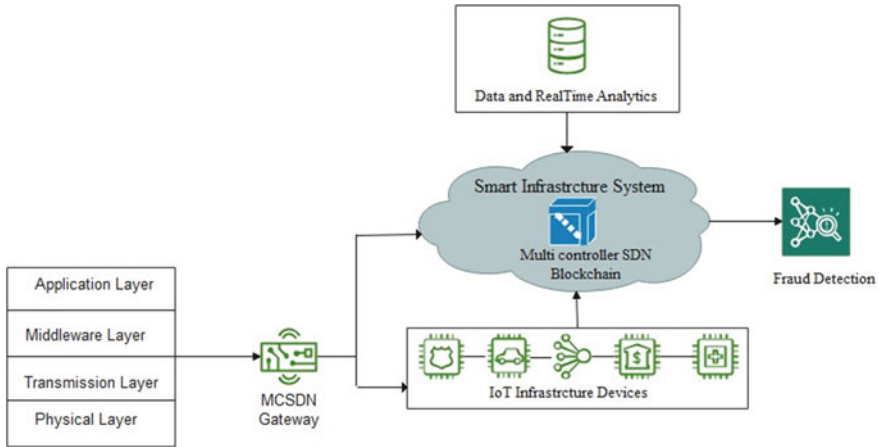


Fig. 1 IoT smart infrastructure layered MCB-SDN architecture

2 Literature Survey

This paper provides a decentralized IoT architecture idea that spans three IoT workflows: computation, storage, and networking in the form of P2P computation overlay, the Ethereum digital signature facilitated decentralized data for IoT entities. As our P2P Storage Overlay, IPFS enabled the widespread storage of IoT data, FL models, and application data. P2P network layers were used to oversee intra-domain and multi-communication using SDN controllers and SDN switches. By adhering to our architecture, we can make IoT computation private without exposing IoT data while preserving reliable IoT storage space and responsive IoT networking. Because we only suggested our architecture concept in this paper, implementing and evaluating it became our urgent future work [1].

To boost security in the cloud storage system, this study presented the Block-SDoTCloud architecture. We also used an SDN infrastructure to direct a distributed blockchain-based process that improved the security, scalability, dependability, confidentiality, and usability of cloud storage services for users. In addition, the writers have successfully performed numerous parameters. By analyzing various procedures, the suggested system provides multiple benefits such as higher throughput, faster response time, faster file transformation, and so on. Furthermore, there are a few restrictions in the proposed system; we did not consider any other assaults in the network layers other than DDoS with flooding attacks. Developers will be able to safely adapt this architectural concept to a variety of applications in the future, including clouds, edge, and mist computing. The system design model will thereafter include further SDN, blockchain, as well as other technologies [2]. In this study, we present a secure network framework that combines three systems: blockchain, SDN, edge, and cloud, for usage in the next phase of IoT ecosystems. The security management framework includes features that are state of the art for next-generation IoT.

The framework, for starters, makes use of blockchain technology. The results show that the proposed security architecture is suitable for fresh research issues in data confidentiality. As a result of the early identification of security breaches, there is less storage required and less delay, as well as a reduction in IoT resource usage and communication bandwidth use. Blockchain technology allows traveling IoT devices and the SDN server to communicate data. Finally, our findings suggest that the suggested security framework be implemented within the IoT network as a data confidentiality preserving element that detects and mitigates any single or collaborative security assaults by monitoring and researching the entire IoT device's traffic flow data.

In recent times, researchers and business verticals have become interested in IoT and IoT big data. While these two technologies actively make people's lives better, they also introduce new threat vectors for future cyber-attacks. IoT networks are an asset and highly heterogeneous when compared to conventional networks. Traditional security measures are inadequate for the IoT context due to these characteristics, necessitating an infrastructure, scalable, and effective security augmentation solution. We begin by examining the characteristics of IoT big data and possible security threats. Then, we present MC-SDNBC, an ID-based SDN security architecture. In this structure, we demonstrated the accuracy of intrusion detection and also overall performance in the presence and absence of our proposed network security using an SDN-specific dataset that models a real IoT environment and contains data recorded for common data assaults and also networks traffic. Our future research will entail expanding the dataset with new attack types and network topologies, as well as evaluating the proposed security strategy under these new network settings. We also want to include an interface for human specialists to interpret the security model, which would improve the model's validity even further [3, 4].

Single point of failure, denial of service attacks, as well as the lack of identification between both the application and the controller were all addressed in this study. We were able to tackle the aforementioned concerns by distributing the SDN control plane across numerous devices while maintaining it logically centralized. Furthermore, blockchain assisted in resolving the common issues that arise when attempting to employ a multicontroller architecture, like device-to-device state synchronization workload is distributed evenly among all processors. A database containing flow entries cannot be changed. For vulnerability analysis and analysis, a record of neural impulses is kept [9]. This research helps ease protection doubts against SDN and encourages industrial adoption of this technology by network engineers by proposing a solution to the security problems discovered in SDN utilizing blockchain. A topology finding mechanism could be added to the smart contract to advance this research. Devices are now added manually via the immediately respond application. Network switches, on the other hand, could be expected to access themselves, as well as the details of surrounding switches, to a list of linked switches in the smart contract when they link to a control layer. The switch can then be approved and added to the topology by the application layer. In the long run, the difficulty bomb function outlined in the previous section might cause mining blocks to take longer and longer to mine, potentially leading to a phenomenon known as the ice age. Even if after Ethereum uses the proof of stake method, this will no longer be an issue [5].

The smart infrastructure network and sensor devices connected in the building need to be more secure to monitor the infrastructures like roads, banks, hospital, buildings, fire service, power supply system, traffic management, gas supply system, homes, digital library, conference hall, etc., the backbone of the smart infrastructure in the ICT transaction with smartly creating physical infrastructure. This ICT infrastructure has a communication protocol like Wi-Fi, fiber optics, hotspot as service-oriented information system [6]. The smart infrastructure is highly efficient, safe and fault-tolerant, and secure as considered to high-level infrastructure which are all physical infrastructure hardware, software, middleware as its overall components. Suppose there is a lot of energy consumption, high maintenance costs, and many abnormal situations [8]. This means this ICT communication gives better ideas and gives solutions to management immediately reflected in smart cities. The use of IoT devices gives an integrated solution that can work and identify the huge amount of data which will higher the operational and power consumption of smart infrastructure (SI). The advantage of SI following: high efficiency, decision making, low-cost operation, more resource gathering, less capital and operational cost structure and management, and risk identification and sustainability [8].

Smart environment monitor system using wireless communication network of ZigBee IoT protocol collects the complete real-time environment information, and here, they started basic monitoring system network connected the street lights as route and taxi's as a node, next dynamically assigned the network every node is allotted with an address as only one identity in the network [10]. The computer design management simulation result is true and can meet the gathered information to structure the terminal in the form of a transaction according to the settings. The multiple sensors added from various intranet devices support multi-functional smart cities based on streetlight and taxis [11]. The multi wireless sensor network model designed with multiple nodes perform different kind of function of the node. Every node divided into a base station, cluster headers, and bash nodes as per their capabilities, which give facilitate an organized and group of the nodes [12]. The hybrid blockchain model is proposed here, connected multi-WSN network model far better, according to various capabilities and energy of various nodes, private and public blockchain delivered in-between cluster header and base station like hybrid network model structured [13].

This paper provides a generic classification of IoT attacks in the latest papers based upon IoT privacy and security, using this technology increased data transaction and networking over the Internet. As per new state-of-art software-based managed devices is called software defined network (SDN) which can fluctuate to conduct a customer's necessary. This attempts to give the taxonomy of previous IoT security threats, and their answers are SDN using the deep learning algorithms [14]. This is also suggested as the primary task of an IoT system to collect data from the devices which is classified into three categories: IoT wireless network, authentication, data aggregation, and validation here remove the cross-layer malicious attack, Bayesian algorithm data validity, neural network are used deep learning [15].

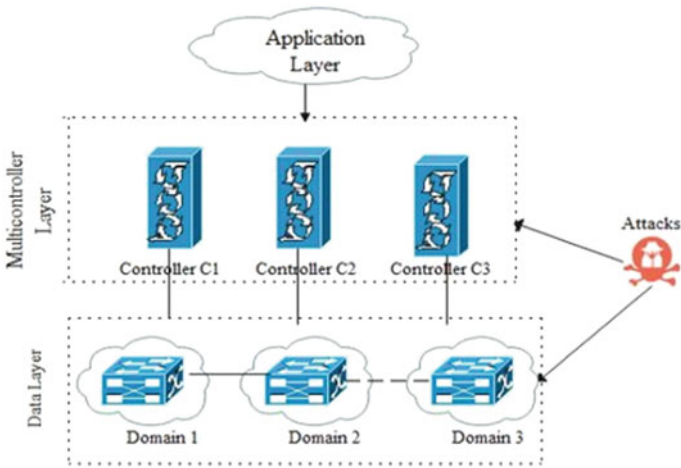


Fig. 2 General multicontroller SDN system

3 Proposed Method

Multiple and dispersed controllers in an SDN network. The application, control, and data layers make up the overall SDN controller. The application layer is made up of programs that tell the actuators about their network architecture and source nodes’ regulations. The control layer is made up of N control systems that are spread over the network. Devices established in N separate domains are found in the data layer. One master controller controls every domain, but every controller unit includes multiple child or duplicate controllers. The controller unit serves as a duplicate controller for multiple domains in addition to its main function. In a distributed system, the controllers. The global view of the network is maintained by multicontroller SDN (Fig. 2). MC-SDN [16] is proposed to manage large-scale and multidomain systems, with each operator accountable with one domain. There have been two types of techniques in MC-SDN: vertical and horizontal. The Openflow [17] handles the southbound connection between both the controller and forwarding devices, such as switches, in verbal leadership by informing switching devices where to get off. The device’s communication with the apps is managed by the network layer. Controllers transmit network information topology via their east–west connections in information exchange.

The network manager and network software’s key concern is keeping the SDN controllers synced and shared significant network information to make the best routing informed choices. Microcontroller SDN, but on the other hand, might be vulnerable to a variety of vulnerabilities, involving false data insertion, in which a hacked controller provides fake flows to other controllers. To address this problem, we offer a security infrastructure that combines blockchain with MCB-SDN (Fig. 3).

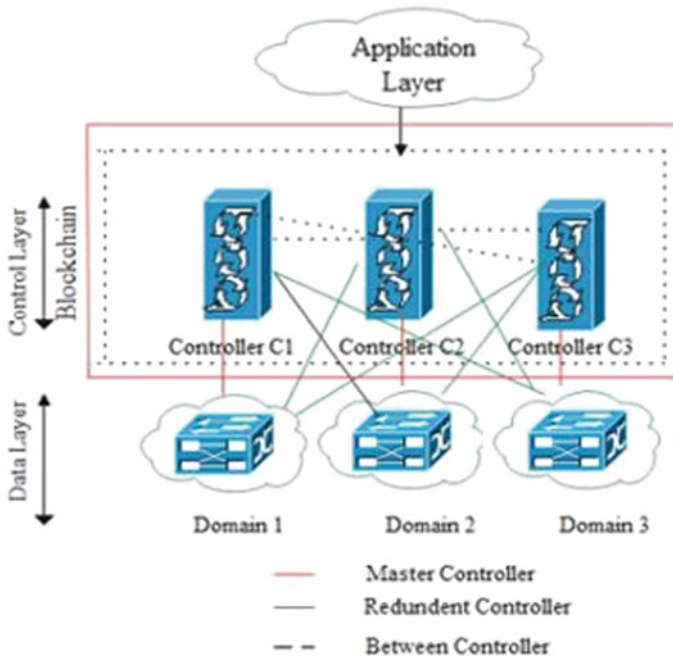


Fig. 3 Proposed MCB-SDN system

The architecture’s core concept is to assign a collection of actuators to each domain. Unlike [18], which uses a large number of controllers for high availability, our design is focused on guaranteeing safe and reliable inter-controller interaction. The proposed framework includes a controller unit and multiple controllers for each virtual network to achieve this goal. Inside one domain, every controller could be the owner, because in other domains, it can be duplicated. The controller unit generates blocks of dynamic network changes, and the duplicate devices decide to choose whether or not authenticate them. The design also includes a popularity system that uses continuous and adaptive fading reputre algorithms to rate the controllers during each voting activity.

4 Methodology

The proposed MC-SDNBC structure is defined in detail in section. The goal of MC-SDNBC is to defend that SDN controller of the previously mentioned multiSDN architecture. In the face of the many vulnerabilities mentioned in Sect. 3, BMC-SDN leverages blockchain to safeguard controller interaction in this way. The control

layer is safeguarded by blockchain. All devices are users of a public blockchain, and devices interact with one another through this network. At MC-SDNBC, we place a premium on information security. The control layer traffic directed towards east-west interface. We evaluate our research in [9] for the integrity of interaction between sensors and control layer components. The number of controllers within the system is denoted by N . We choose a central server controller and M redundant units for each domain, $2 < N \leq M$. Inside the event that the controller unit fails, the duplicate controllers take over. If it is the only redundant regulator available, a duplicate controller cannot substitute several parent controllers. The duplicate controller which will take over the role of a control system is chosen based on its characteristics. The redundant control system with the shortest ID is chosen more accurately. Furthermore, M duplicates controllers in the same database monitor the respective master device's behavior and contribute to the consensus of evaluating the master device's blocks of data.

4.1 *Trusted MCB-SDN Node*

In MCB-SDN, the authorized node has written and read on the blockchain, privileges. All parent operators are regarded as trustworthy data. They will understand and develop new blocks from blockchain adding a new external element to the equation the data layer's message triggers the creation of a new block. When a control board gets new information from its own property's data layer controllers, such as a based on flow notification, it builds a new block having sufficient information and distributes it to the redundant processors for confirmation. All managers in the network have access to the approved block. As a result, each microcontroller can create a global network model that is identical. The duplicate managers are in charge of the consensus process.

- (a) **Trust Multicontroller** if R_i is less than 0.8. The miners assess and take into account the data sent by the controller in this situation.
- (b) **Uncertainty Multicontroller** if $R_i = 0.8$ and 0.4. The evidence provided by the controller is analyzed in this situation; however, the miners do not consider that (Fig. 4).

```

janani@ubuntu:~$ python3 thread.py 10 1 100
flow rules injection at : 25/03/2021 08:22:00
flow rules injection at : 25/03/2021 08:22:01
flow rules injection at : 25/03/2021 08:22:02
flow rules injection at : 25/03/2021 08:22:03
flow rules injection at : 25/03/2021 08:22:04
flow rules injection at : 25/03/2021 08:22:05
flow rules injection at : 25/03/2021 08:22:06
flow rules injection at : 25/03/2021 08:22:07
flow rules injection at : 25/03/2021 08:22:08
flow rules injection at : 25/03/2021 08:22:09

```

Fig. 4 Attack experiment

4.2 Reputation and Consensus MCB-SDN Mechanism

The controllers of this group are known as miners. They are in charge of ensuring that freshly produced blocks are valid. The latest defective block is distributed to the miners once the controller unit introduces a new block. The miners begin the system testing by analyzing the outcomes included in the faulty block to their personal information. The miners get the same application as the control system and respond with the required information. They may, for example, create the same flow rule in response to a certain flow rule request. As a result, the miner may compare the two blocks and approve the new one appropriately after it has been validated, the new node will be uploaded to the blockchain. Malicious controllers could include miners who disagree with the consensus and the control board whose block has still not been confirmed. The following popularity technique can be used to calculate the recognition of the rogue controller. The reputation theory is modeled as such an added step of defense for the SDN controller [19], so the overall system. This strategy is centered on the management of controller reputation. Every controller (C_i) must have a reputation (R_i) value, which is distributed through the chain by all miners. Reputation (R_i) is a number that ranges from 0 to 1 ($0 \leq R_i \leq 1$). Every controller in this system can be in one of three states, based on its reputation score R_i .

4.3 Attack MultiController

If R_i is < 0.4 , this microcontroller's communication traffic is disregarded by the until managed services intervenes, and others will be affected. SDN controller (C_i) reputation is regularly updated when R_i (0:5), and then when R_i 0:4 and R_i 0:8, it transitions to a doubtful and reliable state, accordingly.

4.4 The Consensus C_i Is Evaluated by the Miner Controllers Based on the Consensus Outcome

If a consensus is established, the master device's block will be validated, and also its reputation score may rise. If a consensus cannot be established, the master device's block will not be confirmed, reducing the value of its reputation. The reputation of miners that share the majority opinion viewpoint will improve. Miners whose views differ from the majority will also have their image tarnished [20].

4.5 The Amount of R_i Is Calculated in the Following Way

Throughout each time frame, we calculate the reputation of regulator C_i (R_{Pi}) (or observation interval). R_{Pi} is defined as P_i/T_{Pi} , with P_i is a lot of quality participations made by manager C_i in blockchain activities and T_{Pi} seems to be the overall lots of successful participations made by control C_i (creation and validation of blocks).

4.6 Both Good and Negative Memories Are Remembered at the Same Pace When the Fixed Fading Factor Is Used. Let Us Have a Look at This Link Scenario

If the controller is reliable and then begins to act deliberately, the positive experience will be gradually lost, and the controller's detection rate will indeed belong. If the microcontroller is also not malicious and starts behaving well, the unfavorable past will eventually be forgotten, and the controller's redemption time would belong. If the controller is reliable and then begins to act deliberately, the positive experience will be swiftly forgotten, and the device's detection rate will indeed be short. If indeed the device is malicious then begins to behave well, the negative past will be swiftly forgotten, and also, the controller's redemption time will indeed be quick [21]. Throughout this case, the control system might take advantage of the consensus mechanism and behave maliciously also for the duration of the season, and once the situation of the smart contract becomes suspect or malicious, it will be terminated. The controller would be able to take action. We can see by the examples above that employing a fixed fading factor has various drawbacks [22]. To address this problem, we propose employing varying fading factors based on the controller's trustworthiness.

5 Results and Discussions

IoT Infrastructure for Implementation we use the following elements to construct the blockchain-based secure multicontroller architecture in this section (Fig. 6).

- a. **SDN Control:** The SDN controller is implemented with the open network operating system. It provides the control plane that allows a domain to be deployed with many controllers. The number of SDN domains is 3, the number of duplicate controllers is 2, the switching frequency is (10–100), and the number of connections is (10–450).
- b. **Blockchain:** Multi-communication BC is a technology that allows you to store information to construct a private blockchain, and we use multichannel, an open-source platform.

It can regulate who can connect, transmit, and receive transactions, as well as create flows and blocks by assigning rights to nodes. The multichain Web sample, a basic Web application for multichain blockchains, is used to view each distributed consensus node [23].

- c. **Mininet:** It generates a wireless machine on a single computer that supports OpenFlow and consists of switches and actual apps. It contains the source code which we used to develop MCB-SDN [20]. It generates a wireless machine on a single computer that supports OpenFlow and consists of switches and actual apps. It contains the code which we deployed MCB-SDN to implement. In furthermore, we use postman and other tools to develop our strategy, an option that enables you to submit and handle HTTP requests. SecureCRT, a network management and end-user access software, is used. Our approach is based on Python and certain libraries such as HTTP BasicAuth and Requests, which identify and communicate with the REST APIs for SDN ONOS devices as needed. JSON can be used to serve data that has been handled also by control [21]. Data Structure: The most essential ONOS Stores are ONOS control systems that have used data stores as their true shared data structure. The entire network keep store is among the shared stores, which includes the flow database and the host warehouse. The remaining distributed stores are categorized as software [24, 25] (Figs. 5 and 6).

5.1 The Performance Calculations Are Used to Assess BMC-Performance SDNs in This Category

- a. **Execution Time:** It denotes by (T_{Total}) the amount of time it takes to move a circulation on the blockchain. It is the whole of three factors linked to the number of hosts and switches inside the system: (1) consensus time, (2) block sending time, and (3) information transfer time.

$$\text{TimeTotal} = \text{TimeConsensus} + \text{TimeSent} + \text{TimeUpdate} \quad (1)$$

```
change in the flows detected at : 4779.450109651
beginning flows consensus at : 7.435599945893046e-05
flows consensus not reached : 0.014997593999396486
change in the flows detected at : 4780.477349598
beginning flows consensus at : 0.00041047700051422
flows consensus not reached : 0.011957839000388049
change in the flows detected at : 4781.512537601
beginning flows consensus at : 0.0019234029996368918
flows consensus not reached : 0.01648936000037793
change in the flows detected at : 4782.447263833
beginning flows consensus at : 6.89349999447586e-05
flows consensus not reached : 0.00740328000141603
change in the flows detected at : 4783.471580368
beginning flows consensus at : 7.303100028366316e-05
flows consensus not reached : 0.006153847999485151
change in the flows detected at : 4784.499386722
beginning flows consensus at : 5.969700032437686e-05
flows consensus not reached : 0.009625300000152492
change in the flows detected at : 4785.525112233
beginning flows consensus at : 0.0003000189999511349
flows consensus not reached : 0.008857314999659138
change in the flows detected at : 4786.464522001
beginning flows consensus at : 8.203200013667811e-05
flows consensus not reached : 0.008540415000425128
change in the flows detected at : 4787.491404742
beginning flows consensus at : 0.0008176770006684819
flows consensus not reached : 0.00939876600023262
change in the flows detected at : 4788.525482673
beginning flows consensus at : 0.0042841750000661705
flows consensus not reached : 0.012104711000574753
```

Fig. 5 Thread detection experiment

Fig. 6 Implementation of MC-SDNBC architecture

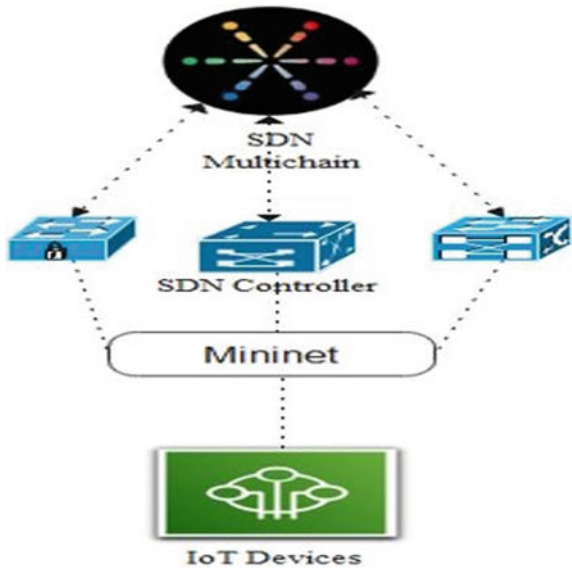


Table 1 No of attack versus DR

Total no of attacks	DR (%)
10	100
20	100
30	100
40	100
50	100
60	100
70	100
80	100
90	100
100	100

- b. **Detection Rate (DR):** This is the number of threats multiplied by the number of attacks. **Detection Time (DT):** It keeps track of how long it takes to detect rogue controllers. **Detection Time(DT):** It keeps track of how long it takes to detect rogue controllers. We insert false flows to the regulator to test the robustness of our MC-SDN method, as portrayed inflows are identified as malicious in Figs. 4 and 5 and notified to the admin by creating a record to the logs giving information of the identified anomaly. Table 1 shows the prediction accuracy versus the number of injected threats. As seen in Tables 2, 3 and Figs. 7, 8 MC-SDNBC provides a detection rate of 100%, meaning that all injected threats were effectively recognized in the system. The duplicate devices have seen the same Internet also as a control system, and the fake flow supplied also by masters is detected by the duplicates during block authentication. We can see that as the switching frequency grows the total runtime grows.

We also notice that as the switching frequency and hosts increase, so does the time it takes to reach a consensus. Despite this, the processing times measured are incredibly short. The proposed system's (Fig. 9) detecting time if a device acts deliberately under three different fading ratios = 0:4; 0:3; 0:8, and the combination fading component where 3 = 0:8, 2 = 0:6, and 1 = 0:3. We could see that the operator's repute declines slowly with a high constant fading rate, resulting in a long detection time (i.e., = 0:8), and rapidly with a that instead of fading factor, resulting in a short detection rate (i.e., = 0:3). We also see that based on the controller's reputation, the total fading factor uses various fading rates. If R_i 0:8 and the fading ratio is large (i.e., = 0:8), the fading component slowly diminishes. If R_i is 0:8, it declines at a faster rate, resulting in a shorter trace level.

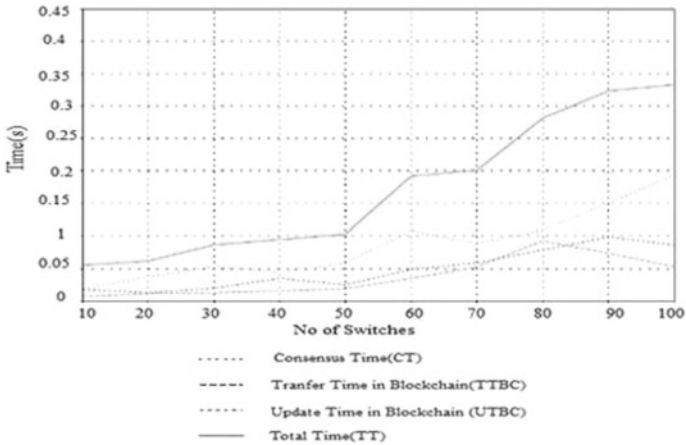


Fig. 7 Number of switches versus execution time

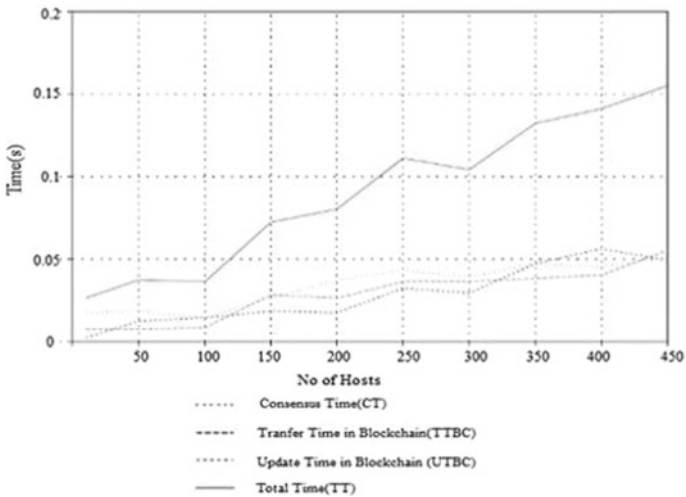


Fig. 8 No of hosts versus execution time

6 Conclusion and Future Work

For secure software-defined networks, MC-SDNBC is a blockchain-based multi-controller design. We cluster wireless networks into SDN domains in this design. Every SDN domain has one master controller and several backup devices. We were using a blockchain, in which the controller unit makes blocks of dynamic network updates, which are then validated by alternative supervisors. Each SDN domain has that there is single master regulator plus several redundant controllers in this system. We were using a blockchain, where the controller unit creates sets of dynamic

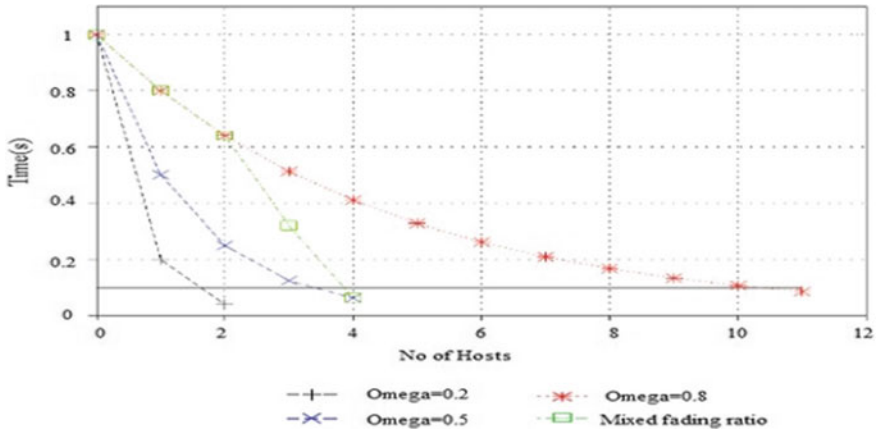


Fig. 9 Detection time of reputation mechanism

Table 2 No of switches versus execution time

Total no of switches	CT	TTBC	UTBC	TT
10	0.018	0.06	0.017	0.054
20	0.037	0.011	0.012	0.061
30	0.053	0.012	0.01	0.086
40	0.043	0.015	0.034	0.094
50	0.058	0.018	0.024	0.102
60	0.017	0.034	0.048	0.191
70	0.088	0.052	0.048	0.201
80	0.108	0.092	0.078	0.282
90	0.15	0.074	0.099	0.323
100	0.193	0.053	0.087	0.333

changes that are subsequently verified by redundant control systems. The controller, block producers, and voters are all rated using a reputé approach, during each voting activity. To monitor and adjust the time consumption of rogue operators, the reputation system combines constant and dynamic combined fading reputation algorithms. ONOS, multi-blockchain, and Mininet software platforms have all been used to construct and test the proposed security IoT architecture. In a short period, the evaluation findings showed that flow rule injections were detected 100% of the time. Furthermore, dynamic fading factor adjustment was facilitated by the obtained with the proposed reputation system to reach the required detection time. Because MC-SDNBC only looks at the integrity of east–west interconnections, we aim to address the remainder of the security layers of SDN architecture in future work, particularly the southbound interfaces.

Table 3 No of hosts versus execution time

Total no of hosts	CT	TTBC	UTBC	TT
10	0.018	0.008	0.003	0.027
50	0.019	0.008	0.013	0.038
100	0.015	0.009	0.015	0.035
150	0.027	0.029	0.019	0.073
200	0.038	0.027	0.018	0.09
250	0.044	0.037	0.033	0.111
300	0.039	0.036	0.029	0.104
350	0.047	0.038	0.047	0.132
400	0.046	0.05	0.056	0.141
450	0.051	0.055	0.049	0.155

References

- Liu M, Song T (2019) Deep cognitive perspective: resource allocation for NOMA-based heterogeneous IoT with imperfect SIC. *IEEE Internet Things J* 6(2)
- Rahman A, Islam J (2020) Block-SDoTCloud: enhancing security of cloud storage through blockchain-based SDN in IoT network
- Medhane DV, Sangaiah AK (2020) Blockchain-enabled distributed security framework for next generation IoT: an edge-cloud and software defined network integrated approach. *IEEE Internet Things J*
- Sarica AK, Angin P (2020) Explainable security in SDN-based IoT networks
- Krishnamohan T, Janarthanan K (2020) BlockFlow: a decentralized SDN controller using blockchain. *Int J Sci Res Publ* 10(3)
- Lv Z, Hu B (2020) Infrastructure monitoring and operation for smart cities based on IoT system. *IEEE Trans Ind Inform* 16(3)
- Mohanty SP, Choppali U (2016) Everything you wanted to know about smart cities. *IEEE Consum Electron Mag*
- Yazdinejad A, Parizi RM (2020) An energy-efficient SDN controller architecture for IoT networks with blockchain-based security. *IEEE Trans Serv Comput*
- Wani A, Revathi S (2021) SDN-based intrusion detection system for IoT using deep learning classifier (IDSIoT-SDL). <https://doi.org/10.1049/cit2.12003>.
- Bhayo J, Hameed S (2020) An efficient counter-based DDoS attack detection framework leveraging software defined IoT (SD-IoT)
- Lounis K, Zulkernine M (2020) Attacks and defenses in short-range wireless technologies for IoT
- Frustaci M, Pace P (2018) Evaluating critical security issues of the IoT world: present and future challenges. *IEEE Internet Things J* 5(4)
- Hu T, Guo Z (2018) Multi-controller based software-defined networking: a survey. *IEEE*
- SDN controller. <https://github.com/dariobanfi/multipath-sdn-controller>
- Mininet. <https://github.com/mininet/mininet>
- Hu T, Guo Z (2018) Multi-controller based software-defined networking: a survey
- Rajabi N, Qaddour J (2019) SDIoBoT: a software-defined internet of blockchains of things model. <https://doi.org/10.5923/j.ijit.20190801.03>
- Karmakar KK, Varadharajan V (2020) SDN enabled secure IoT architecture. *IEEE Internet Things J*. <https://doi.org/10.1109/JIOT.2020.3043740>

19. Li W, Meng W (2020) Towards blockchain-based software-defined networking: security challenges and solutions. *IEICE Trans Inf Syst* E103–D(2)
20. Thorat P, Dubey NK (2021) SDN-based predictive alarm manager for security attacks detection at the IoT gateways. In: 2021 IEEE 18th annual consumer communications & networking conference (CCNC)
21. Blial O, Mamoun MB (2016) An overview on SDN architectures with multiple controllers. <https://doi.org/10.1155/2016/9396525>
22. Ramya G, Manoharan R (2021) Enhanced optimal placements of multi-controllers in SDN. <https://doi.org/10.1007/s12652-020-02554-2>
23. <https://thenewstack.io/multiple-sdn-controllers/>
24. <https://github.com/knetsolutions/learn-sdn-with-ryu/blob/master/overview.md>
25. <https://sdnwiselab.github.io/>

A Recent Survey on Cybercrime and Its Defensive Mechanism



Garima Bajaj, Saurabh Tailwal, and Anupama Mishra

Abstract Cybercrime is one of the severe issues in today's world that is increasing day by day due to unawareness of people about the harm it can cause. The main reason against the augmentation of cybercrime is the lack of education or knowledge about the impact it can lead to. Cybercrime can be done against individuals, society, or any organization whether it is private or government. The aim of the paper is to focus on what cybercrime is, its types, related work, and its defensive mechanism. Defensive mechanism against cybercrime includes the ways or measures that how can any individual or any organization protects them against cybercrime. This paper also includes the related work which includes some points about the work which has been done so far on cybercrime.

Keywords Cyber security · Cyber crime · Phishing · Cyber stalking · Social engineering

1 Introduction

The word cybercrime is the combination of two words cyber + crime which means the crime related to computer or the things related to it. Cybercrime is the crime that can be caused by the involvement of computer, network, or any network device. Cybercrime can be carried out by a particular individual or by big organization. A hacker hacking any person's data or the data of any organization for his/her profit is also a part of cybercrime. Stealing someone's pictures without his/her permission and using it badly just in order to satisfy revenge is also a part of cybercrime. There are tremendous amount of examples of cybercrimes in India. In India, cybercrime is increasing day by day because the use of Internet or you can say the number of Internet users is increasing day by day due to lack of understanding about cybercrime or about the impact it can cause. The misuse of Internet is spreading in India which

G. Bajaj (✉) · S. Tailwal · A. Mishra
Swami Rama Himalayan University, Dehradun, India
e-mail: garimabajaj9818@gmail.com

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
D. P. Agrawal et al. (eds.), *Cyber Security, Privacy and Networking*, Lecture Notes
in Networks and Systems 370, https://doi.org/10.1007/978-981-16-8664-1_29

339

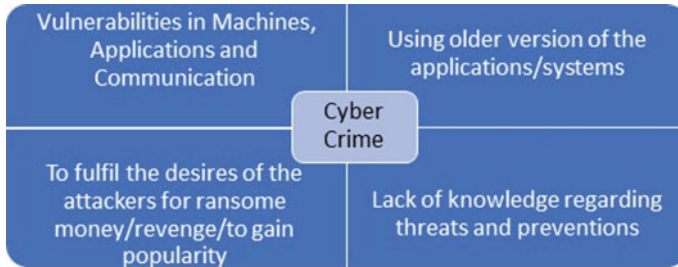


Fig. 1 Causes of cybercrime

in turns increases the rate of cybercrime. Our daily lives are replete with information technologies that we rely on to simplify our lives. In today's environment, mobile phones, the Internet, and email have become indispensable for communication. Every common man is familiar with the terms "hacker" and "virus," which are frequently used in conjunction with data loss, sophisticated theft of money, and compromised security. Cybercrime is becoming increasingly prevalent these days [1] (Fig. 1).

2 Related Work

Cybercrime is defined as crimes committed using a communication channel or device, whether it is a laptop, desktop, PDA, mobile phone, watch, or vehicle, directly or indirectly. According to the report, titled "Global Risks for 2012," cyberattacks will be one of the top five risks to the world's governments and businesses in 2012. Cybercrime is a type of crime that is more difficult to detect and harder to stop once it has occurred, resulting in long-term negative consequences for victims [2]. Although the concept of cybercrime is not new, there is considerable confusion among academics, computer security experts, and users regarding the scope of true cybercrime. We examine the breadth of computer-based crime in this article, including a definition of the emerging terms "cybercrime" and "crimeware." Then, we categorize cybercrime into two distinct categories: Type I Cybercrime, which is primarily technological in nature, and Type II Cybercrime, which has a stronger human component. Then, using two case studies, we demonstrate the role of crimeware in various types of cybercrime and make some observations about the role of cognition in the cybercrime process [3].

3 Classification of Cybercrimes

Cybercrimes can be categorized based on many perception like based on criminal behaviour, based on crime type, based on size of the target like individual/organization/society.

3.1 *Cybercrimes Based on Target Size*

Cybercrimes are divided into three main groups.

- (1) Cybercrime against individuals—The type of cybercrime which is done against a particular person or against people. It includes:
 - Email harassment
 - Spoofing with email
 - Cyberstalking
 - Unauthorized access
 - Fraud.
- (2) Cybercrime against organization—The type of cybercrime which is done against any organization whether it is government or private or any other company. It includes:
 - Retrieval of unauthorized information
 - Cyber terrorism
 - Hacking of organization's server
 - Distribution of pirated software.
- (3) Cybercrime against society—The type of cybercrime which is done against society. It includes:
 - Trafficking
 - Gambling
 - Forgery
 - Spoiling the youth with filthy material.

3.2 *Cybercrimes Based on Crime*

- (1) Forgery—When an offender alters documents stored in a computerized form, the crime associated with it is known as forgery. In this computer systems are the target to carry out such criminal activities.

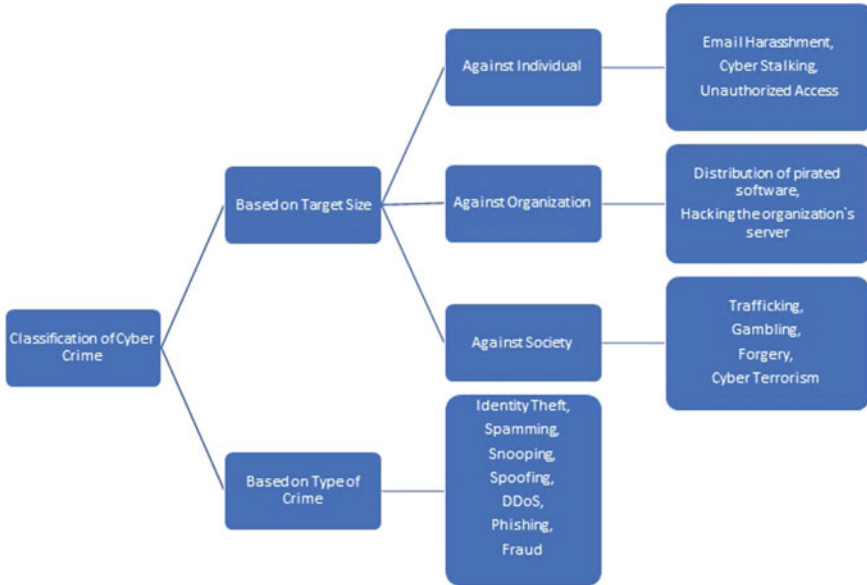


Fig. 2 Classification of cybercrime

- (2) Email spoofing—It is referred to as crime in which sender sends the fake mails to the receiver. In this origin, details have been altered so as to make it appear that it originates from other source. It is used to launch phishing attacks [4].
- (3) Cyberstalking—It refers to the use of email, Internet to commit criminal activity which includes harassment of victims without the victim's permission, and in this way, the criminal can create fear in victim. Cyberstalking is ignited by rage, power, control, and anger that have been triggered by victim's action or in many cases victim's inactions [5].
- (4) Hacking—It is the art of solving problem in a creative manner that means finding an uncommon solution to a hard problem or manipulating holes in an unsystematic programming [6].
- (5) Cyber terrorism—Cyber terrorism is defined as the use of computer network tools to close down critical national infrastructures such as government operations [7].
- (6) Phishing—Phishing is a form of attack in which an attacker aims to acquire sensitive information from a victim by portraying itself as a reliable entity [8].
- (7) Fraud—Fraud is increasing worldwide with the increase in use of modern technology resulting in the loss of billion dollars throughout the world. Fraud includes credit card fraud, telecommunication fraud, etc. [9].
- (8) Gambling—Gambling can occur almost in all cultures and in every period of time. It refers to risk taking activities. It can be understood as staking of money, investment in stock market. It allows an individual or group of organizations to extract profit [10] (Fig. 2).

4 Defensive Mechanisms Against Cybercrime

4.1 Information Assurance

To protect our information, there are five basic principles:

- (1) Confidentiality—Confidentiality means to keep the information or data confidentiality and can only be accessible to authorized users. Only authorized users can copy and use that information. For example, you give permission to someone for viewing the information who is not authorized, but the authorized user is allowed to completely access it.
- (2) Integrity—Data integrity means to maintain the integrity of data where an unauthorized user is not allowed to alter or delete data. Data integrity can occur when our computer is attacked by the virus or when hacker gains unauthorized access to our server and can delete and modify our important data.
- (3) Authenticity—It means that the user should be an authorized person who has his own credentials like username and password and documents of the users cannot be altered without user's permission.
- (4) Availability—Availability means that the information should be available to the authorized users and the measures to protect the file can be properly taken to protect it and will make sure that the exact information should be available to a correct person.
- (5) Non-repudiation—Non-repudiation means the guarantee that the person cannot deny the validity of something. It assures that there are enough evidences that the person cannot deny over something.

4.2 To Implement Defense in Deep Plan of Action

- To broaden organizational boundaries: Businesses today maintain tight ties with their business partners, consumers, and suppliers. This results in difficult-to-define exterior boundaries; for example, when business partners join an association for the purpose of delivering a product, it is because they share the same infrastructure, computer systems, and personnel. It is critical to define the organization's boundaries and how it is implementing its defense in depth plan.
- Mobile workforce: It is critical for employees to be capable of accessing their company's network from a remote place. Employees must have access to the same software applications and data as those at the corporate headquarters. This interconnection enables viruses and hackers to spread throughout the system, causing damage.
- Decentralization of services: As there is increase in the use of computers at workplace, on the other hand, there is increase in the services. Earlier these services are provided to a limited amount of users, but now, they are provided to a broad

category of users. As from business point of view, the business information is very important, so it will be a prime responsibility to protect this information from unauthorized access, and in this way, it may assist in achieving good governance and improves in delivery of service.

The steps for implementing defense in deep plan of action

- **Internal and external environment analysis:** This is the foremost step for implementing defense in deep plan of action. It is very important for an organization to check internal and external environment in which it is operating. It should be very important for an organization to know about its strengths and weakness and what are the threats an organization will going to face and what technology and processes are being used. It is also very important for an organization to properly strategize about defense plan in action and clearly understands about the steps to implement defense in deep plan of action.
- **Determining the risks:** This is the second step which determines the risks which an organization will face. It includes the threats and vulnerabilities. It is necessary for an organization to identify risk and will take proper measures to reduce these risks. An organization should always be ready for such risks and with proper mitigation steps.
- **Strategy of defense in-depth implementation:** If all risks have been identified properly, then now it comes to deal with such risks with proper pre-planned strategy and will make use of proper defense mechanism.
- **Maintenance, monitoring, and review:** As we all know that technology is increasing day by day so the risk of threats and risks are also increasing so it very important to properly monitor, maintain, and review all this and will adapt the changes accordingly [11].

4.3 Education

It is one of the most important defensive mechanisms. People should be educated about the harmful impact of cybercrime. They should be aware of the punishments which are there under the IT Act about the offence off cybercrime committed by them. They should be aware of the do's and don't over the Internet. People should know that following someone and using other people's private information is also a part of cybercrime (Fig. 3).

5 Conclusion

The objective of this paper is to spread awareness about what cybercrime is and what are its types. In this paper, we have studied about existing definition and work about cybercrime, and following this, we have added definition of cybercrime according

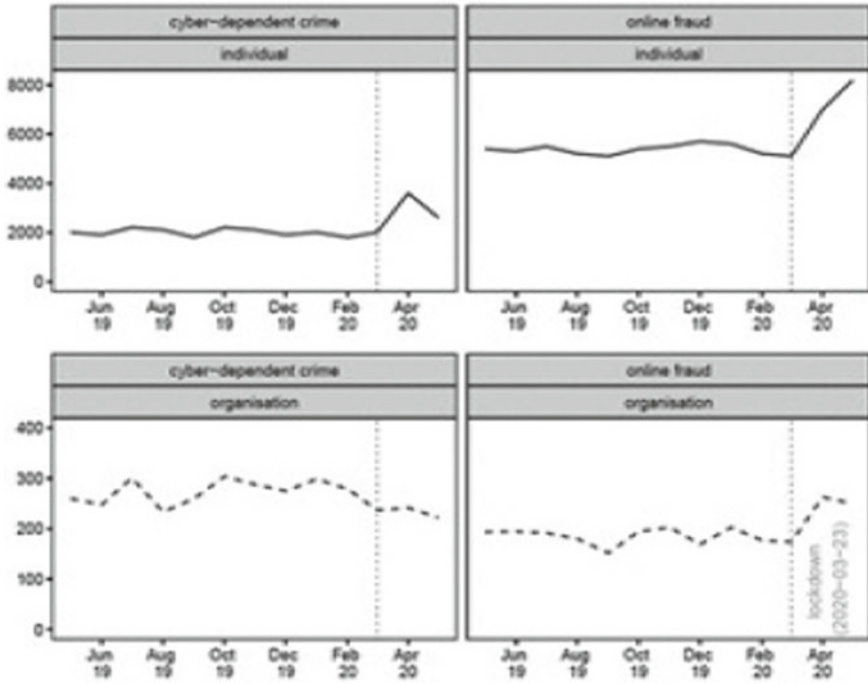


Fig. 3 Cybercrime statistics

to our understanding. Types of cybercrimes have been also added based on present scenarios. With the help of our increased understanding about the types of it, many people can be solved from becoming a part of cybercrime. Finally, we conclude that understanding about cybercrime is very important and spreading awareness about the impact of it is very important. By studying about the defensive mechanism of cybercrime, we can reduce the rate of cybercrime.

References

1. Furnell S (2002) Cybercrime: vandalizing the information society. Addison-Wesley, London, pp 3–540
2. Gunjan VK, Kumar A, Avdhanam S (2013) A survey of cyber crime in India. In: 2013 15th international conference on advanced computing technologies (ICACT), Sept 2013. IEEE, pp 1–6
3. Gordon S, Ford R (2006) On the definition and classification of cybercrime. *J Comput Virol* 2(1):13–20
4. Pandove K, Jindal A, Kumar R (2010) Email spoofing. *Int J Comput Appl* 5(1):27–30
5. Pittaro ML (2007) Cyber stalking: an analysis of online harassment and intimidation. *Int J Cyber Criminol* 1(2):180–197
6. Erickson J (2008) Hacking: the art of exploitation. No Starch Press

7. Lewis JA (2002) Assessing the risks of cyber terrorism, cyber war and other cyber threats. Center for Strategic & International Studies, Washington, DC, p 12
8. Jagatic TN, Johnson NA, Jakobsson M, Menczer F (2007) Social phishing. *Commun ACM* 50(10):94–100
9. Bolton RJ, Hand DJ (2002) Statistical fraud detection: a review. *Stat Sci* 17(3):235–255
10. McMillen J (1996) Understanding gambling. In: *Gambling cultures: studies in history and interpretation*, pp 6–42
11. Fick J (2009) Prevention is better than prosecution: deepening the defence against cyber crime. *J Digit Forensics Secur Law* 4(4):3
12. Buil-Gil D, Miró-Llinares F, Moneva A, Kemp S, Díaz-Castaño N (2020) Cybercrime and shifts in opportunities during COVID-19: a preliminary analysis in the UK. *Eur Soc* 1–13
13. Mishra A, Gupta N, Gupta BB (2020) Security threats and recent countermeasures in cloud computing. In: *Modern principles, practices, and algorithms for cloud security*. IGI Global, pp 145–161
14. Mishra A, Gupta N (2019) Analysis of cloud computing vulnerability against DDoS. In: *2019 international conference on innovative sustainable computational technologies (CISCT)*, Oct 2019. IEEE, pp 1–6
15. Bhushan K, Gupta BB (2019) Distributed denial of service (DDoS) attack mitigation in software defined network (SDN)-based cloud computing environment. *J Ambient Intell Humaniz Comput* 10(5):1985–1997
16. Alsmirat MA et al (2019) Impact of digital fingerprint image quality on the fingerprint recognition accuracy. *Multimed Tools Appl* 78(3):3649–3688
17. Dahiya A, Gupta BB (2020) Multi attribute auction based incentivized solution against DDoS attacks. *Comput Secur* 92:101763
18. Al-Qerem A et al (2020) IoT transaction processing through cooperative concurrency control on fog cloud computing environment. *Soft Comput* 24(8):5695–5711
19. Gupta S, Gupta BB (2015) PHP-sensor: a prototype method to discover workflow violation and XSS vulnerabilities in PHP web applications. In: *Proceedings of the 12th ACM international conference on computing frontiers*, May 2015, pp 1–8
20. Dahiya A, Gupta BB (2020) A reputation score policy and Bayesian game theory based incentivised mechanism for DDoS attacks mitigation and cyber defense. *Future Gener Comput Syst*

A Hybrid Feature Selection Approach-Based Android Malware Detection Framework Using Machine Learning Techniques



Santosh K. Smmarwar , Govind P. Gupta , and Sanjay Kumar 

Abstract With more popularity and advancement in Internet-based services, the use of the Android smartphone has been increasing very rapidly. The tremendous popularity of using the Android operating system has attracted malware attacks on these devices. Detecting variants of malware features that change their behavior to hide from being detected by the traditional method of machine learning is being an incapable and challenging task. To overcome these issues of malware feature detection, an efficient feature selection plays a crucial role in detecting malware features and reduces the dimensionality of a huge dataset and removes the unnecessary features that are not useful and keeps those relevant features that improve the classification accuracy and detection rate. To address the above issues, this paper proposed a novel framework in which a hybrid feature selection using wrapping feature selection (WFS) with the combination of random forest and greedy stepwise (RF-GreedySW) framework is devised to optimize the malware features. The proposed framework is capable of reducing a large number of attributes into an optimal feature to enhance the performance of the machine learning model. The framework used the three most popular ML classifiers such as random forest (RF), decision tree (C5.0), and support vector machine radial basis function (SVM RBF). The performance of the proposed framework is evaluated using the CIC-InvesAndMal2019 dataset. The DT (C5.0), RF, and SVM RBF model achieves better accuracy of 91.80%, 91.32%, and 82.33% on static layer, respectively. Similarly, the accuracy is 72.41%, 75.10%, and 62.07% on the dynamic layer by DT (C5.0), RF, and SVM RBF, respectively. Our model highlights good results on the CIC-InvesAndMal2019 dataset in terms of classification accuracy and increases the robustness of the model.

Keywords Machine learning · Random forest · Wrapper feature selection · Android malware detection · Ransomware · Adware · API calls

S. K. Smmarwar (✉) · G. P. Gupta · S. Kumar
Department of Information Technology, National Institute of Technology Raipur, Raipur, India
e-mail: santoshsmmarwar@gmail.com

S. Kumar
e-mail: skumar.it@nitrr.ac.in

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
D. P. Agrawal et al. (eds.), *Cyber Security, Privacy and Networking*, Lecture Notes in Networks and Systems 370, https://doi.org/10.1007/978-981-16-8664-1_30

347

1 Introduction

The rapid growth in the commercialization of Android platforms, digital services, the huge number of online service availability, and connectivity in smart devices have raised cyber-threat to user's privacy and security. These arises security concerns to the device's data privacy, integrity, and confidentiality. The attacker compromises the loopholes by installing malicious programs, uses them to access the sensitive information from the user's system. In recent times, there are more than 5 billion mobile customers as well as around 12 billion Internet of things devices are being used [1]. The increasing number of online services has attracted the threat of malware attacks. Malware is a software code having bad intension regarding the system resources, data collection, modification of codes, disguise users from normal activities for financial benefits, etc. Malware does unauthorized activities to steal valuable information, slows down the system process, consumes device memory, and sometimes demands money. There are various kinds of malware classes such as viruses, worms, Trojans, adware, spyware, Ransomware, SMSware, and many more exist [2]. Malware attacker uses evasion techniques by making the new variants of malware class to bypass the detection by using the obfuscation techniques. Two common methods used in malware analysis that is the static analysis and dynamic analysis. In static analysis, the malware is detected without running the codes. However, the static analysis is not effective to detect mutant malware [3, 4]. Some of the previous studies show that static approaches are weak in detecting new variants of malware. Instead of using static approaches, the dynamic method is capable at some level to detect the obfuscated file having a malicious nature in the virtual environment [5, 6]. However, these existing studies used the approaches like machine learning and deep learning shows some limitations like lower detection rate of malware and their category, classification accuracy, selecting the most suitable feature to predict malware [7–10]. So, in this work, we have proposed the wrapping feature selection (WFS) framework for selecting optimal features by using random forest and the greedy stepwise (RF-GreedySW) search method. The following are the main contribution of this research works as follows.

1. Proposed a novel malware detection framework in which a novel hybrid feature selection approach by combining the basic wrapping method with random forest and greedy stepwise (RF-GreedySW) search method is devised to optimize the malware features.
2. For detection of the malware, three ML classifiers such as random forest (RF), decision tree (C5.0), and support vector machine radial basis function (SVM RBF) are used.
3. Performance evaluation of the proposed framework is evaluated using the CIC-InvesAndMal2019 dataset in terms of accuracy and detection rate.

The remaining part of this paper consists of the following sections below, Sect. 2 is the related work of Android malware detection, Sect. 3 is the proposed framework,

Sect. 4 is the analysis and discussion of the results, and finally, Sect. 5 is the conclusion of the work.

2 Related Work

This section presents work related to Android malware detection approaches used in the previous studies. In an Android operating system, malware detection has done mainly based on three features like permission, intents, and API calls. The effectiveness of a malware detection system depends on the important attributes to detect efficiently variants of malware. In [11], the author worked on the detection rate of Ransomware by using a machine learning classifier from the Android-based dataset CICAndMal2017 of ten Ransomware families. The CICAndMal2017 dataset contains benign and malware applications [12] and consists of four types of malware categories as Adware, Ransomware, Scareware, and SMS Malware. In paper [13], the CICAndMal2017 dataset related to a single PCAP file was used for each malware family randomly. Similarly, in [14], authors have developed the lightweight detection system for the static feature by using the latent semantic indexing approach provides a reduced set of features to improve the detection rate. This lightweight detection system is evaluated on a machine learning classifier in which a random forest classifier is well performed. However, this work is done only for the static feature that limits the performance of the model.

3 Hybrid Feature Selection Approach-Based Android Malware Detection Framework

Here, we have proposed the hybrid feature selection approach-based Android malware detection framework. This framework used the wrapping feature selection (WFS) approach using the random forest and greedy stepwise (RF-GreedySW) search method to optimize the malware features. The dataset of CIC-InvesAndMal2019 contains the static feature and dynamic feature of malware. The static layer includes permission and intents feature, while the dynamic layer feature consists of API calls and other log files. Static layer samples contain the benign application data, and a malware category sample includes adware, premium SMS, Ransomware, scareware, and SMS malware. The dynamic layer contains malware samples such as Ransomware, scareware, SMS malware, and Adware. Figure 1 shows that the proposed wrapper feature selection framework consists of preprocessing phase, model training, and finally, the malware classification phase for malware detection, and a brief explanation of each phase is given below.

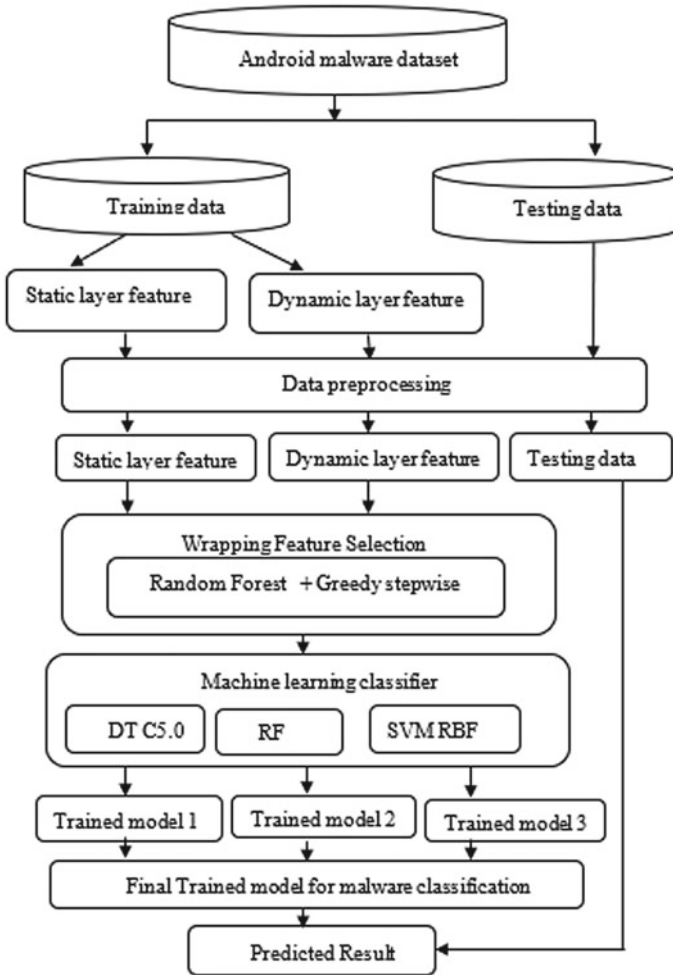


Fig. 1 Proposed framework for Android malware classification

3.1 Data Preprocessing

Preprocessing of data is the essential step to make data in a standard form for machine learning models to work well in classification. Original data is transformed into a required format, removes the missing values, and changes header name to prevent the misleading of the result. Therefore, it is necessary to transform data before going to data analysis. In our work, we removed the missing data, renaming of header name.

3.2 *Wrapping Approach*

The wrapping technique is used to select the best subset of features from the large number of features set using the machine learning algorithm. The wrapping approach utilized the search strategy to find a subset of features from the space vector of the feature set, and these check each selected subset based on the performance of the algorithm. The learning algorithm selects the subset of features in such a way that the obtained features are smaller than an original feature, thus provided better performance capability to the model and gives good predictive accuracy. In wrapping, we used the random forest for subset evaluator and greedy stepwise work in both directions forward or backward to get the optimal subset.

1. **Random Forest:** Random forest is an attribute evaluator and selects a subset of attributes sets using learning schemes. The cross-validation used to estimate the accuracy of the learning scheme for a set of attributes.
2. **Greedy Stepwise:** The greedy stepwise is an attribute selection algorithm and works as a greedy forward or backward search through the space of attribute subsets. It starts with selecting no/all attributes or from an arbitrary point in the space and stops working when the addition or deletion of any remaining attributes results in a decrease in evaluation. This can also produce a ranked list of attributes by traversing the space from one side to the other and recording the order that attributes are selected.

3.3 *Machine Learning Algorithm*

This section discussed some of the basic machine learning classifiers that were employed on the Android dataset to measure the performance of our approach as well as accuracy.

- (a) **Decision Tree (C5.0):** This is the classification model of supervised learning used to create a binary tree or multi-branches tree. It was developed in the year 1994 by Ross Quinlan used the information gain or entropy for data splitting. C5.0 is used to solve various kinds of problems by using the automatic learning process to tackle the numeric, nominal, and missing values, provide the best result by partitioning the dataset into small subparts. It is useful for high-dimensional datasets to predict relevant and irrelevant features for classification purposes.
- (b) **Random Forest (RF):** Random forest algorithm is the most efficient supervised learning classifier to predict the accurate result. It generates multiple decision trees by using bootstrap samples in resampling training data and follows the ensemble learning approach to handle the complex and difficult problems for improving the prediction accuracy of the model. The ensemble learning approach combines the weak learner into the strong learner.

- (c) **Support Vector Machine (SVM RBF)**: SVM is a state-of-the-art classification model, used the RBF as a computational high power kernel-based tool for classification. It is used in various areas due to its high accuracy capability and handles high-dimensional data. SVM aims to maximize the hyperplane so that more features are separated. The kernel function used hyperparameters known as gamma and regularization parameters. The gamma values are used to improve the accuracy of the model, and the regularization value reduces the misclassification of data points.

4 Result Analysis and Discussion

The performance evaluation of our proposed framework is done on the CIC-InvesAndMal2019 Android dataset. The work is classified into two parts for the classification of Android malware that is on a static layer and dynamic layer.

4.1 *Experimental Setup and Evaluation Parameter*

In this work, the proposed framework used the Java-based environment Weka 3.8.4 tool for feature selection and optimization. The experiment was performed on Windows 10 with a configuration of Intel core i3-2330 processor 2.20 GHz with 8 GB RAM and using the R tool. The performance parameter and experimental setup have the main role to analyze the effectiveness of the machine learning model. We have taken datasets for training and testing in the ratio of 80:20, respectively, and calculated the accuracy, sensitivity, specificity, kappa statistics, and AUC-ROC values for evaluation of our framework as mentioned in [15, 16].

4.2 *Static Layer Malware Category Detection*

Table 1 shows the accuracy and kappa statistics of different machine learning classifiers evaluated on the CIC-InvesAndMal 2019 dataset. The accuracy obtained by all three classifiers DT, RF, and SVM RBF is 91.80, 91.32, and 82.33%. Among all three classifiers, the best accuracy is obtained by the DT classifier.

The kappa statistics of the machine learning model are used to assess the classification performance of the model. The kappa statistics are computed by all three models as 79.56%, 77.52%, and 50.12% by DT, RF, and SVM, respectively, on the static layer. The AUC-ROC curve is 0.95, 0.93, and 0.90 of ML models as shown in Fig. 2 of DT, RF, and SVM, respectively, indicating the better performance of the model. This shows the significant improvement in the overall performance of the malware detection rate.

Table 1 Comparison of accuracy and kappa statistics on static layer for malware category classification

ML classifier	Accuracy (%)	Kappa statistics (%)
DT (C5.0)	91.80	79.56
RF	91.32	77.52
SVM RBF	82.33	50.12

Highest result shown in bold values

Table 2 Comparison of sensitivity and specificity on static layer

Malware category	DT (C5.0)		RF		SVM RBF	
	Sensitivity	Specificity	Sensitivity	Specificity	Sensitivity	Specificity
Adware	71.05	98.82	68.42	99.16	44.73	98.15
Benign	98.10	79.37	97.47	80.63	98.73	53.75
PremiumSMS	90.00	99.83	95.0	99.18	95.00	99.83
Ransomware	89.18	99.16	86.48	98.49	10.05	100
Scareware	56.08	99.32	56.09	99.15	04.87	98.65
SMS malware	66.66	99.67	58.33	99.50	66.66	97.04

Table 2 demonstrated the sensitivity and specificity of a state-of-the-art machine learning classifier with optimizing the feature of the android dataset on the static layer. The sensitivity values of malware range 56.08–98.10% for DT, 56.09–97.47% for RF, and 04.87–98.73% for SVM RBF. The specificity values of the malware class are 79.37–99.83% for DT (C5.0), 80.63–99.50% for RF, and 53.75–100% for SVM RBF.

4.3 Dynamic Layer Malware Category Detection

Table 3 demonstrated an accuracy and kappa statistics comparison of three ML models are evaluated on the CIC-InvesAndMal2019 dataset. The accuracy achieved by these models is 72.41%, 75.10%, and 62.07 by DT, RF, and SVM RBF, respectively, on tenfold cross-validation, and the highest accuracy is achieved by RF models.

The kappa statistics of ML models in Table 3 is to be computed as 62.92% is highest for DT (C5.0), 61.64% of RF, and 44.38% of SVM RBF. Figures 2 and 3 represent the ROC comparison chart of tenfold CV models for all models. The ROC curve of each model is plotted simultaneously. Area under the curve (AUC) measures the area under an entire ROC curve. If the value of AUC-ROC is found greater than 0.5, a model is considered better and appropriate for developing a prediction model. The AUC-ROC value of the three ML classifiers comes out to be 0.97 for RF, 0.99 for DT, and 0.71 for SVM RBF. The AUC-ROC value of the DT model is 0.99 which is

Table 3 Comparisons of accuracy and kappa statistics on the dynamic layer for malware category classification

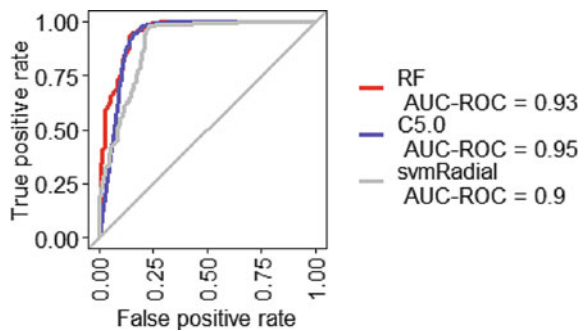
ML classifier	Accuracy (%)	Kappa statistics
DT (C5.0)	72.41	62.92
RF	75.10	66.26
SVM RBF	62.07	50.32

Highest result shown in bold values

Table 4 Comparison of sensitivity and specificity on the dynamic layer

Malware category	DT (C5.0)		RF		SVM RBF	
	Sensitivity	Specificity	Sensitivity	Specificity	Sensitivity	Specificity
Adware	69.57	88.17	60.87	92.47	78.26	68.82
Ransomware	83.33	93.48	79.17	91.30	70.83	90.22
Scareware	59.38	91.67	65.62	84.52	18.75	92.85
SMSmalware	78.38	89.87	78.38	93.67	70.27	93.67

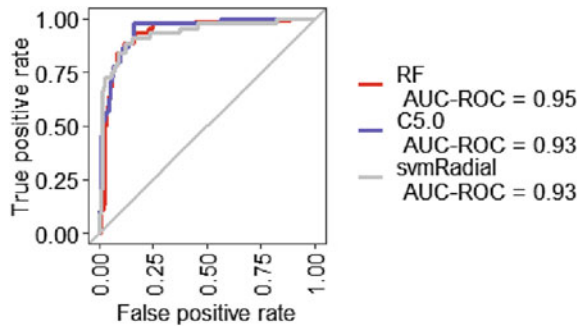
Fig. 2 ROC curve for tenfold cross-validation on static layer



far greater than 0.5 implies that the proposed model including other models is good to build a prediction model and not fall under random guesser.

The results from Table 4 contain the comparison of sensitivity and specificity values on the dynamic layer. Sensitivity values for adware, Ransomware, scareware, and SMS malware are to be computed by three machine learning models to test the performance of the model. The highest sensitivity value is 83.33%, and the lowest is 59.38% for the Ransomware malware by DT (C5.0) model as compared to other classifiers (RF, SVM RBF). The sensitivity values of the RF model for Ransomware are 78.38% which is the highest and 60.87% is the lowest. The sensitivity value of another classifier by SVM RBF of 78.26% is the highest for adware, and 18.75% is the lowest for scareware.

Fig. 3 ROC curve for tenfold cross-validation on the dynamic layer



5 Conclusion

This research work proposed a novel malware detection framework in which a novel hybrid feature selection approach by combining the wrapping method with random forest and greedy stepwise (RF-GreedySW) search method is devised to optimize the malware features. Our study uses the most popular machine learning models such as DT (C5.0), RF, and SVM RBF to identify malware types using the latest Android dataset known as CIC-InvesAnd2019. The potential application of our approach can be in the problems like object identification and image segmentation where feature selection is a challenging task. From the above result, we can be concluded that our proposed framework is effective and efficient in malware detection. In the future, we plan to implement our framework based on deep learning techniques using different real-time datasets.

References

1. Imtiaz SI, ur Rehman S, Javed AR, Jalil Z, Liu X, Alnumay WS (2021) DeepAMD: detection and identification of Android malware using high-efficient Deep Artificial Neural Network. *Future Gener Comput Syst* 115:844–856
2. Vasan D, Alazab M, Wassan S, Naeem H, Safaei B, Zheng Q (2020) IMCFN: image-based malware classification using fine-tuned convolutional neural network architecture. *Comput Netw* 171:107138
3. Venkatraman S, Alazab M (2018) Use of data visualization for zero-day malware detection. *Secur Commun Netw*
4. Shafiq M, Tian Z, Bashir AK, Du X, Guizani M (2020) IoT malicious traffic identification using wrapper-based feature selection mechanisms. *Comput Secur* 94:101863
5. Alzaylaee MK, Yerima SY, Sezer S (2020) DL-Droid: deep learning-based android malware detection using real devices. *Comput Secur* 89:101663
6. D'Angelo G, Palmieri F, Robustelli A, Castiglione A (2021) Effective classification of Android malware families through dynamic features and neural networks. *Conn Sci* 1–16
7. Tchakounté F, Djakene Wandala A, Tiguiane Y (2019) Detection of Android malware based on sequence alignment of permissions. *Int J Comput (IJC)* 35(1):26–36

8. Yuan Z, Lu Y, Xue Y (2016) DroidDetector: Android malware characterization and detection using deep learning. *Tsinghua Sci Technol* 21(1):114–123
9. Yuan Z, Lu Y, Wang Z, Xue Y (2014) Droid-Sec: deep learning in android malware detection. In: *Proceedings of the 2014 ACM conference on SIGCOMM*, Aug 2014, pp 371–372
10. Jerbi M, Dagdia ZC, Bechikh S, Said LB (2020) On the use of artificial malicious patterns for android malware detection. *Comput Secur* 92:101743
11. Noorbehbahani, F., Rasouli, F., & Saberi, M. (2019, August). Analysis of machine learning techniques for ransomware detection. In *2019 16th International ISC (Iranian Society of Cryptology) Conference on Information Security and Cryptology (ISCISC)* (pp. 128-133). IEEE
12. Android malware dataset (CIC-AndMal2017) (2021) <https://www.unb.ca/cic/datasets/andmal2017.html>. Accessed 2021-05-11
13. Chen R, Li Y, Fang W (2019) Android malware identification based on traffic analysis. In: *International conference on artificial intelligence and security*, July 2019. Springer, Cham, pp 293–303
14. Singh AK, Wadhwa G, Ahuja M, Soni K, Sharma K (2020) Android malware detection using LSI-based reduced opcode feature vector. *Procedia Comput Sci* 173:291–298
15. Kumar P, Gupta GP, Tripathi R (2021) Toward design of an intelligent cyber attack detection system using hybrid feature reduced approach for IoT networks. *Arab J Sci Eng* 46(3):3749–3778
16. Kumar P, Gupta GP, Tripathi R (2021) Design of anomaly-based intrusion detection system using fog computing for IoT network. *Autom Control Comput Sci* 55(2):137–147

Security of Big Data: Threats and Different Approaches Towards Big Data Security



Yashi Chaudhary and Heman Pathak

Abstract In the present era, the use of the Internet has extended abruptly. With this abrupt increase, massive data is being created, resulting in big data. Big data means more diverse, more impetus, and more complex data streams. Data is being produced in abundance in exabytes and zettabytes by electronic devices, power grids, and modern software. This big data brings different challenges such as incompleteness, inconsistency, heterogeneity, and security with itself. The presented paper targets the security challenge as it is a very significant feature overseen by various data analysts; thus, data must be secured from dwindling in the wrong hands. This paper discusses the approaches and mechanisms mainly based on anonymization, access control, and encryption.

Keywords Access control · Anonymization big data · Big data life cycle · Big data security · Encryption · High volume · Storage · Introduction

1 Introduction

The term **big data** is used to define humongous data that could either be of structured, semi-structured, or unstructured type. The extensively large data makes processing difficult by using traditional available databases and software technologies. Heavy parallel software devices running on thousands of servers can be used for processing [1].

Big data initially was categorized by the four Vs—volume, variety, velocity, and veracity. However, with time other categorizations are also made as the data is emerging vastly with each passing day [2, 3].

Y. Chaudhary (✉) · H. Pathak
Gurukul Kangri University, Haridwar, India
e-mail: mohita.chaudhary5@gmail.com

1.1 *Ten Vs of Big Data*

Big data is mainly characterized by the Vs that define the different traits of the data one is dealing with the under mentioned figure and describes significant ten Vs that make big data different from the traditional data.

Volume. Generated data's quantity is referred to as volume. Value and potential of the data are examined with the size of the data and whether it can be put into the category of big data or not it is also examined through the size of the available data. The huge scale increment makes the analysis of data a difficult process if one is using traditional available tools.

Variety. The category of the data it belongs to is known as variety. Data can come in any form. It may be structured data, unstructured data, or semi-structured data coming out of various sources like e-mails, videos, audios, transactions, etc.

Velocity. How fast the data is generated and processed to meet the required demand refers to as velocity. The speed with which information is gathered and processed to meet the required demands of the intended users.

Veracity. It can be referred to as the trustworthiness of the data that is being used. Analysis correctness depends heavily on the veracity of the source data. Data captured quality can differ immensely.

Variability. It refers to different things. It focuses on adequately understanding and interpreting the correct meaning of raw data that depends on its context. It can also be defined as the inconsistency of the speed at which data is generated and stored. Validity means how accurate the data is for its specified use. If one wants to use the results in some decision-making, subsequent analysis must be accurate enough.

Vulnerability. Vulnerability means a flaw that can leave a system open to attack. The vulnerability may also be referred to as any type of lapse in a computer system, in a set of procedures, or in anything that can hinder the security of the system.

Volatility. In the world of real-time analysis, it is important for the decision-makers to analyse till when the data provided is relevant. This relevancy of data validity is known as volatility.

Visualization. Data visualization means how the data is presented in a graphical format that is easily understood and interpreted by its users. Various complex representations like heat maps and fever charts are included here that help decision-makers to identify the hidden patterns and correlations.

Value. It is the most important trait of the data. Without this, other characteristics are of no use if we are not able to deduce the business value from the data. Big data helps in decision-making in the organization by measuring the importance of the data.

1.2 Big Data Analytics

Big data analytics means exploring large data sets containing a variety of datatypes in big data—to reveal hidden data patterns, unknown interactions, market trends, customer preferences, and other useful business information. To make companies more knowledgeable by enabling data scientists is the first goal of big data analysis. Big data comprises structured, semi-structured, and unstructured data. Tools that are used for advanced analytics such as predictive analysis, data mining, and text analysis can be used in big data analysis as well. Data visualization tools along with some mainstream BI tools can also be very effective in the analysis process of big data [4].

Big data analytics life cycle—As we are using vast data repositories to gain information that will be useful for analytics purposes, we need to refine the available data. The refinement process includes various steps as defined in Fig. 1.

In all the above phases, there are multiple threats that are required to take care of.

Data Collector. Data comes from various sources and with different formats, i.e. structured, semi-structured, and unstructured. In this phase, information is gathered to address various things that can be used by an organization for various purposes. From the security point of view, securing big data from the first phase is very important. Limited access control and encryption of data fields can be done here to ensure privacy here [5].

Data Storage. Data storage mainly addresses the volume challenge by making use of distributed, shared nothing architectures. Data is stored and prepared here that will be used in the next phase. Here produced data may be sensitive, so it is vital to take care of it. Data anonymization, permutation, data partitioning, etc., are some techniques that can be applied to ensure security [6].

Data Analytics. The primary aim of big data analysis process is to disintegrate the significant data from the bunch of data and to provide decisions and recommendations based on the findings after investigating the whole data. This phase is used to create

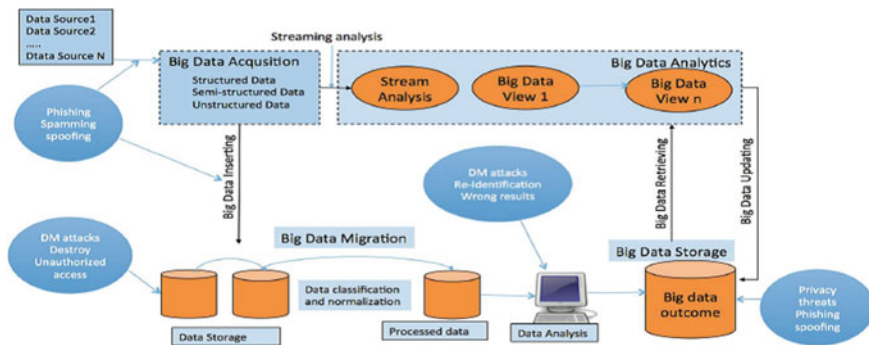


Fig. 1 Phases of big data analytics

Table 1 Threats on various phases of big data analytics

Phase	Threats	Description
Data collector	Spoofing	These attacks are performed to get access to the data collection phase
	Spamming	
	Phishing	
Data storage	Data mining-based attacks	Targets the data sets to extract knowledge
	Attacks on data storage devices	These include stealing the hard discs
	Unauthorized data access	People access the data illegally
Data analytics	Data mining attacks	Uses data mining methods to extract valuable information
	Re-identification attacks	Includes personal threat identification
Knowledge creation	Privacy threats	Releasing the resulted knowledge
	Phishing	Decision-makers are targeted
	Spoofing	Decision-makers are targeted

knowledge. Various data mining methods can be used here. Data miners use powerful algorithms that can extract sensitive data. A security breach may also happen here [7].

Knowledge Creation. This is the final phase. Conversion of the data into some useful information is done at this step. If data seizing and sensing are done right, then big data repositories can be created in the form of knowledge repositories. It is used by decision-makers. New information and valued information are created here. Knowledge is sometimes considered sensitive here [8].

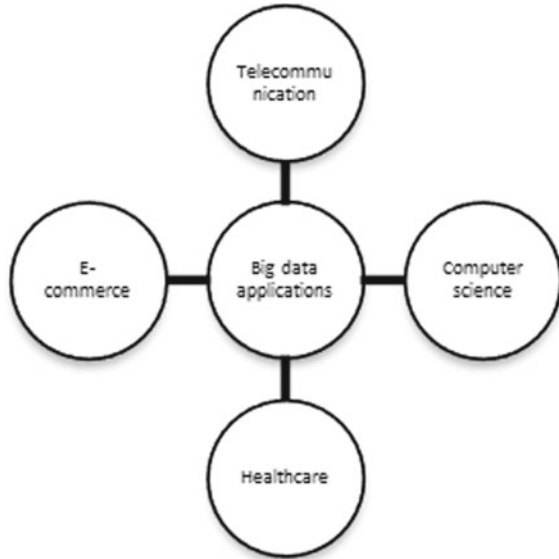
Threats Associated with Big Data Life Cycle

Various threats associated with different phases of the big data life cycle have been summarized in Table 1 [2].

1.3 Applications of Big Data

In the present era, the use of the Internet has extended abruptly. Due to the vast usage of the Internet anywhere and everywhere, big data applications are also increased due to their decision-making ability. Big data is no more just a buzzing word, but its use is everywhere today. All credit goes to the technology that is nowadays not

Fig. 2 Various applications of big data



just confined to the urban areas, but rural and underdeveloped areas are also taking its advantage. Big data applications range from the water supply, smart cities, crime, health care, education, electricity, etc. (Fig. 2).

1.4 Challenges with Big Data

Generated from different devices at a very fast pace, big data brings the following challenges with itself:

Security. Data is generated at a high pace in huge quantities every day. Big data analytics will not be considered a reliable system if security algorithms will not be taken into account. Security issues can be further categorized: input, analysis of data, and output, system communication.

Inconsistent Data. More inconsistent data and incompatible data will easily appear since data is being gathered from different systems. So it will also be a challenge while doing big data analytics.

Privacy. It is different from the issue of security as it deals with the fact that whether it is possible to restore the personal information of the system with the help of big data analytics, even though the input variables are anonymous. With big data analytics being widely used, it is quite possible that private information may get exposed to other people after the analysis process. So it is also a challenge in big data analytics.

Heterogeneity. Insights of data can be achieved through the richness and nuances of the data. Though, machine algorithms cannot understand nuances as they expect comparable data. So structuring the data carefully is the first step of big data analysis. Even after applying data cleaning and data correction methods, some incompleteness of data may be there. Managing this is a great challenge.

Timeliness. In a dynamic and rapidly growing world, a second or even a microsecond between one reading and the other may led to mismatching against each other. So timeliness is a very fundamental concept while dealing with real-time data.

Communication Between the Systems. Since most of the tasks of the big data analytics system will be designed for parallel computing, big data analytics and other systems communication will impact the performance of the system immensely. So managing the cost of communication and making connections reliable are two open challenges to deal with [2].

2 Big Data Security: A Multifaceted Challenge

Big data security is a cumulative term for all the techniques and tools that are used in securing the data against all malicious activities such as theft of data, attacks, or any activity that affects negatively. The other threats include DDoS attacks, ransomware, and online stored information stealing [9, 10].

The prime reason for security concerns in big data is because big data can be accessed widely nowadays. Data is shared on a large scale by scientists, doctors, business officials, government agencies, and normal people. The current approaches are inadequate when dealing with big data security. The present technology has weak security capability for maintenance. So intruders can easily breach those. Thus, reassessment and updation of current approaches should be performed to prevent data leakage [11]. There are various challenges when one is dealing with big data security. A few of them are mentioned below.

2.1 Issues

Vulnerability to fake data generation—Before dealing with all the operational security challenges of big data, the concerns of counterfeit data generation should be kept in mind. To purposively undermine the quality of the big data analysis, cybercriminals can forge the data. For instance, if a manufacturing company uses sensor data to detect malfunctioning production processes, cybercriminals can penetrate that system and make sensors show fake results, say, wrong temperatures. This way, one can fail to notice alarming trends and miss the opportunity to solve problems that can cause severe damage. Such issues can be addressed by applying the fraud detection approach [12].

Untrusted mapper's presence—After collection, big data firstly undergoes parallel processing. MapReduce paradigm is used here and when data splits a mapper processes that and allocates a position for storage to the data. If anyone from outside knows your mapper's code, he/she can change it. In this way, it is ruining the information processed very effectively. Outsiders can get inside to access sensitive information [13].

Mining of sensitive information—Perimeter-based security ensures data protection at entry and exit levels. But inside the system, the work of IT professionals is a mystery. Such a lack of control over big data solutions can allow corrupted IT professionals to mine the data and sell it for their benefit. As a result, the organization will suffer huge losses. Here, data can be made more secure by adding values to it. Anonymization can also benefit the system's security. The private details with absent names, telephones, etc., practically will not harm if someone acquires this information with malicious intentions [14].

Real-time protection of data—It is hard for organizations to maintain orderly checks as data is generated vastly on a real-time basis. However, security checks in real time or almost in real time will prove beneficial [15].

Access control granularly—Granular access control allows people to access the required sets of data but can view the only part of data they are allowed to see. The whole valuable content will not be visible to them. Vastly, it can be very useful in health care where sensitive information like names and phone numbers will remain hidden while other information may be useful for medical researchers to find new insights [16].

Privacy protection of non-relational database—Various security vulnerabilities are faced by datastores such as NoSQL that lead to privacy threats. At the time of logging and tagging, it is unable to encrypt the data and so is the case with the distribution of data to different groups while it is streamed and collected [16].

3 Security-Based Literature Survey

There are three major security considerations outline that has been taken into account while dealing with big data: anonymization, encryption, and access control [17] (Fig. 3).

Diversified data sources, data streams, data formats, and infrastructures may impose unique security vulnerabilities (Table 2).

3.1 Existing Approaches to Handle the Big Data Security

Listed below are the different approaches to manage security as discussed by various authors.

Table 2 Summary of the literature survey

S. No.	Authors name	Problem identified by authors	Security algorithm discussed	Security aspects
1.	Li et al.	Low computation time over the cloud	Security-aware efficient distributed storage (SA-EDS) model	Encryption
2.	Aljawarneh et al.	High computational cost	The amalgamation of Feistel encryption scheme, an advanced encryption standard (AES), and genetic algorithms	
3.	Yan et al.	Data deduplication security issue	Proxy re-encryption algorithm	
4.	Dong et al.	Sensitive data	Proxy re-encryption algorithm based on heterogeneous ciphertext	
5.	Hu et al.	Access control of data	ABAC algorithm	Access control
6.	Zeng et al.	High overhead of conventional algorithms while dealing with big data	Content-based access control (CBAC)	
7.	Khuntia et al.	User's private information leakage	Hidden policy-ciphertext policy-attribute-based encryption: HP-CP-ABE	
8.	Siffah et al.	High risk of data leakage	MeDShare: blockchain for sharing trustless medical data	
9.	Jasim et al.	Zero trust between models	Transaction's manager model algorithm	Anonymization
10.	Zhang et al.	Low scalability due to high I/O cost	MONDRIAN WITH MapReduce (MRMONDRIAN)	
11.	Al Zobi et al.	Ignorance of generalizations	MDSBA (expanded K-anonymity algorithm)	
12.	Ferrer et al.	Overlapping populations and increase in quasi-identifiers	Advanced K-anonymity algorithm	
13.	Mehta et al.	Loss of information	Improved scalable l-diversity	
14.	Cui et al.	Confidentiality of shared data	Attribute-based storage system	

Fig. 3 Approaches towards big data security



Security by encryption—Enabling only the authorized user’s access to the information by encoding the information is known as encryption. Li et al. proposed an algorithm to avoid cloud operators reaching the user’s sensitive data. It is the amalgamation of AD2, SED2, and efficient data conflation algorithms entitled as security-aware efficient distributed storage model [18]. Aljawarneh et al. proposed a system for multimedia big data against real-time tampered data attacks. The proposed scheme is made by merging the Feistel network, AES, S-Box, and genetic algorithm. The scheme is applied over the data set of JUST university hospital [19]. Yan et al. proposed a scheme based on deduplication of encrypted data and proxy re-encryption. Deduplication is an important practice to achieve successful cloud storage, especially for big data storage. It allows only the authorized users to access the information. It supports flexible data updates offline as well [20]. Dong et al. presented a scheme for heterogeneous ciphertext transformation. It is a proxy algorithm that works on a virtual-based monitor which provides support for the realization of system functions. It is designed to protect and secure user’s data effectively. It also provides the data owner the total control over their data for modern information security [21].

Security by access control—One of the most important security components is access control systems. Due to misconfiguration of the access control policies, the security and privacy of the system are often compromised. Hu et al. have proposed a scheme for distributed big data processing clusters. The scheme aims to authorize the protection of big data processing from internal attacks [22]. Wnorong et al. have introduced the mechanism for content access. The proposed mechanism is very suitable for the content-sharing of information in big data. CBAC is used for access control decisions based on semantic similarity between the requester’s credentials and the content [15]. Siffah et al. proposed an off-chain-based sovereign blockchain where transactions are made between parties through a virtual container. Then blockchain network is used to store the output [23]. Kumar et al. proposed a scheme based on ciphertext policy with an attribute—encryption along with less computation overhead [24]. Khuntia et al. proposed a scheme for privacy preserving in the cloud to ensure big data access control. To reduce computational overhead, authors have used the concept of multi-sharing here [25]. Jasim et al. proposed a

three-tier approach including cloud architecture, transaction manager, and clients. Zero trust is the basis of communication between the models [26].

Security by anonymization of the data—Control over private information gathering and its usage is information privacy. The ability to stop information from becoming public either by a group or an individual is known as information privacy. The assimilation of private information over the Internet during its transmission is one of the issues faced by the users. Privacy protection is one of the most bothering issues in big data and cloud applications, so there is an urgent need for strong customer privacy preservation techniques. Data anonymization is one of the efficient and effective ways towards privacy preservation [27]. Zhang et al. proposed a technique based on MapReduce on the cloud. A combination of highly scalable median finding algorithm and histogram technique is used here to propose for achieving cost effectiveness. Scalability is also measured here using multivariate partitioning [28]. Zhang et al. have pointed out the scalability issue in the cloud over big data. For this, a hybrid approach of top-down specialization and bottom-up generalization is used. K-anonymity parameter with workload sharing is used for selecting the component to achieve a highly scalable environment if compared with the existing approaches [29]. Al Zobi et al. have proposed a novel framework MDSBA. According to the authors, the loss of important information is the result of the avoidable generalized identical details. Through the proposed scheme, authors have expanded the k-anonymity and applied the bottom-up approach to avoid the identical widespread records more methodically and efficiently [30]. Ferrer et al. have focused their work towards dealing with the two important issues while using k-anonymity, i.e. the quasi-identifier attributes and the data controllers attributes by proposing a k-anonymity algorithm that avoids the dimensionality problem and by using mean and median to avoid the risk of disclosure by replacing the generalization method with the alternative aggregation method which is comparatively less sensitive, respectively [31]. Cui et al. proposed a deduplication-based system for a hybrid cloud used for attribute-based storage. The authors also discuss the ways to achieve semantic security along with keeping in mind the context of confidentiality to share the data with other users [32]. Mehta et al. proposed a scheme with the name improved scalable l-diversity approach based on K-anonymization. The run-time of this scheme is very less, and the loss of information is also less in comparison with other schemes [33].

4 Conclusion

Data is increasing with each passing moment over the Internet, making it impossible for traditional approaches to deal with the data. Out of the available bulky and raw data, extracting the relevant information is the important task of big data analytics. However, while dealing with the data, security is the major threat that is being faced by the analysts. The present paper discusses some of the novel approaches that can be

used to ensure the security of big data. Moreover, we have noted that all the present traditional schemes cannot be applied over big data, but with certain advancements, in the future, the schemes can be improved and applied.

References

1. Arora M, Bahuguna H (2016) Big data security—the big challenge. *Int J Sci Eng Res* 7(12)
2. Tarekegn GB, Munaye YY (2016) Big data: security issues, challenges and future scope. *Int J Comput Eng Technol* 7(4):12–24
3. Siddique M, Mirza MA, Ahmad M, Chaudhry J, Islam R (2018) A survey of big data security solutions in healthcare. In: *International conference on security and privacy in communication systems*, Aug 2018. Springer, Cham, pp 391–406
4. Bhadani AK, Jothimani D (2016) Big data: challenges, opportunities, and realities. In: *Effective big data management and opportunities for implementation*. IGI Global, pp 1–24
5. Elgendy N, Elragal A (2014) Big data analytics: a literature review paper. In: *Industrial conference on data mining*, July 2014. Springer, Cham, pp 214–227
6. Zuech R, Khoshgoftaar TM, Wald R (2015) Intrusion detection and big heterogeneous data: a survey. *J Big Data* 2(1):1–41
7. Ruiz-Rosero J, Ramirez-Gonzalez G, Williams JM, Liu H, Khanna R, Pisharody G (2017) Internet of things: a scientometric review. *Symmetry* 9(12):301
8. Sagiroglu S, Sinanc D (2013) Big data: a review. In: *2013 international conference on collaboration technologies and systems (CTS)*, May 2013. IEEE, pp 42–47
9. Mujawar S, Kulkarni S (2015) Big data: tools and applications. *Int J Comput Appl* 115(23):7–11
10. Joseph Charles P, Carol I, MahaLakshmi S (2018) Big data security—an overview. *IRJET* 11(2)
11. Tarekegn GB, Munaye YY (2016) Big data: security issues, challenges and future scope. *IJCET* 4(7):12–24
12. Sisense. <https://www.sisense.com/glossary/big-data-security>. Accessed 2019/12/03
13. Joseph A, Cherian M (2018) The quest for privacy and security in various big data applications: a survey. *IJCESR* 3(5):1–8
14. Lafuente G (2015) The big data security challenge. *Netw Secur* 2015(1):12–14
15. Begoli E, Horey J (2012) Design principles for effective knowledge discovery from big data. In: *2012 joint working IEEE/IFIP conference on software architecture and European conference on software architecture*, Aug 2012. IEEE, pp 215–218
16. Xu L, Jiang C, Wang J, Yuan J, Ren Y (2014) Information security in big data: privacy and data mining. *IEEE Access* 2:1149–1176
17. Zhang D (2018) Big data security and privacy protection. In: *8th international conference on management and computer science (ICMCS 2018)*, vol 77, Oct 2018. Atlantis Press, pp 275–278
18. Li Y, Gai K, Qiu L, Qiu M, Zhao H (2017) Intelligent cryptography approach for secure distributed big data storage in cloud computing. *Inf Sci* 387:103–115
19. Aljawarneh S, Yassein MB (2017) A resource-efficient encryption algorithm for multimedia big data. *Multimed Tools Appl* 76(21):22703–22724
20. Yan Z, Ding W, Yu X, Zhu H, Deng RH (2016) Deduplication on encrypted big data in cloud. *IEEE Trans Big Data* 2(2):138–150
21. Dong X, Li R, He H, Zhou W, Xue Z, Wu H (2015) Secure sensitive data sharing on a big data platform. *Tsinghua Sci Technol* 20(1):72–80
22. Hu VC, Ferraiolo D, Kuhn R, Friedman AR, Lang AJ, Cogdell MM, Scarfone K (2013) Guide to attribute based access control (ABAC) definition and considerations (draft). *NIST Spec Publ* 800(162):1–54

23. Zeng W, Yang Y, Luo B (2013) Access control for big data using data content. In: 2013 IEEE international conference on big data, Oct 2013. IEEE, pp 45–47
24. Xia QI, Sifah EB, Asamoah KO, Gao J, Du X, Guizani M (2017) MeDShare: Trust-less medical data sharing among cloud service providers via blockchain. *IEEE Access* 5:14757–14767
25. Khuntia S, Kumar PS (2018) New hidden policy CP-ABE for big data access control with privacy-preserving policy in cloud computing. In: 2018 9th international conference on computing, communication and networking technologies (ICCCNT), July 2018. IEEE, pp 1–7
26. Jain P, Gyanchandani M, Khare N (2016) Big data privacy: a technological perspective and review. *J Big Data* 3(1):1–25
27. Jasim AC, Tapus N, Hassoon IA (2018) Access control by signature-keys to provide privacy for cloud and big data. In: 2018 5th international conference on control, decision and information technologies (CoDIT), Apr 2018. IEEE, pp 978–983
28. Zhang X, Qi L, Dou W, He Q, Leckie C, Ramamohanarao K, Salcic Z (2017) Mrmondrian: scalable multidimensional anonymisation for big data privacy preservation. *IEEE Trans Big Data*
29. Zhang X, Liu C, Nepal S, Yang C, Dou W, Chen J (2014) A hybrid approach for scalable sub-tree anonymization over big data using MapReduce on cloud. *J Comput Syst Sci* 80(5):1008–1010
30. Al-Zobbi M, Shahrestani S, Ruan C (2016) Sensitivity-based anonymization of big data. In: IEEE 41st conference on local computer networks workshops (LCN workshops). IEEE, pp 58–64
31. Domingo-Ferrer J, Soria-Comas J (2016) Anonymization in the time of big data. In: International conference on privacy in statistical databases, Sept 2016. Springer, Cham, pp 57–68
32. Cui H, Deng RH, Li Y, Wu G (2017) Attribute-based storage supporting secure deduplication of encrypted data in cloud. *IEEE Trans Big Data* 5(3):330–342
33. Mehta BB, Rao UP (2019) Improved l-diversity: scalable anonymization approach for privacy preserving big data publishing. *J King Saud Univ Comput Inf Sci*

Segmentation of Image Using Hybrid K-means Algorithm



Roopa Kumari and Neena Gupta

Abstract Image segmentation is a crucial step to recognizing an object. During the segmentation process, pixels in an image are categorized based on their gray color. In pixel classifications, the K-means clustering algorithm is commonly used. In this approach, the centroid of the segment was measured using arithmetic mean and Euclidean distance. In the proposed paper, the centroid was updated using the hybridization of harmonic and arithmetic means. The proposed algorithm makes use of the harmonic and arithmetic mean features. The experimental results are compared to conventional K-means and harmonic K-means algorithms, demonstrating that the proposed algorithm performs better when checking segmentation consistency.

Keywords Image segmentation · Harmonic mean · Arithmetic mean · K-means clustering

1 Introduction

Image segmentation is a necessary stage in the image processing. Based on certain characteristics or features, image segmentation divides an image into multiple subparts. We may extract some useful data from an image using segmentation for better understanding or further processing. Image segmentation assigns a class label to each pixel in the image, and pixels with the same label share similar properties. Medical imaging, robotics applications, satellite imaging, agricultural imaging, traffic control systems, and object recognition are few examples of image segmentation applications [1]. Detecting discontinuity or similarity can be used to segment data. For example, in detecting discontinuity, an image is split based on rapid changes in intensity, and in detecting similarity, an image is split based on similarity, as in the region-based process [2].

R. Kumari (✉) · N. Gupta

Department of Computer Science, Gurukul Kangri Deemed to University, Kanya Gurukul Campus, Dehradun, Haridwar, Uttarakhand, India
e-mail: rooparawal@gmail.com

Image segmentation has various methods to partition an image like thresholding, edge-based, region-based, and clustering techniques. All these techniques are conventional. Some soft computing techniques are also used to segment an image like fuzzy-based techniques, particle swarm optimization, artificial neural networks, and evolutionary algorithms [3]. A single segmentation algorithm cannot be used on every type of image, or a single solution cannot be used to solve every problem. As a result, different approaches are used to solve various problems. The simplest and most significant approach for image segmentation is clustering approach. Clustering is the process of grouping data objects based on their similarity into clusters [4]. The clustering algorithm aims to create a partitioning decision using an initial collection of clusters that is revised after each iteration [5].

This paper aims to create a hybrid method of K-means algorithms. The process for K-means, harmonic mean, and arithmetic mean is detailed in Sect. 2. In Sect. 3, the proposed algorithm is described, and in Sect. 4, the experimental result of the proposed algorithm is described, along with comparisons to the K-means and harmonic K-means algorithms. The conclusion of the paper is presented in Sect. 5.

2 Related Work

2.1 *K-means Clustering Algorithm*

K-means clustering is a method to group a set of data into a specific number of groups or clusters. K-means is one of the most useful and easy methods of clustering technique which was developed by Macqueen in 1967 [6]. Clustering techniques are commonly used in various fields such as AI, machine learning, data compression, data mining, marketing, and medicine according to rapidly rising data [7]. K-means method divides a collection of data into K -number groups or clusters [8] and similarity led to the formation of these K clusters. The “Euclidean distance measuring function” is the most commonly used function to calculate similarity. Clusters will be pre-defined. The numbers of pre-defined clusters are denoted by the letter K . It is an iterative approach that divides an unlabelled data set into K clusters based on similarities, with each data set belonging to only one cluster. The K-means clustering technique consists of two phases or modes: (1) An iterative method is used to measure the K centroid. (2) Assign each data point to the K centroid closest to it. The K-means algorithm, while being used in a wide range of applications, yet has some flaws that K-means method is extremely sensitive to the initial starting conditions (initial clusters and instance order). To overcome this problem, various approaches have been proposed, but there is always a trade-off between efficiency and precision. The basic K-means mechanism is depicted in Fig. 1.

The K-means method is a type of evolutionary method that gets its name from the way it works. Because of its benefits, such as ease of implementation and speedy convergence, K-means is more common. However, K-means has some drawbacks,

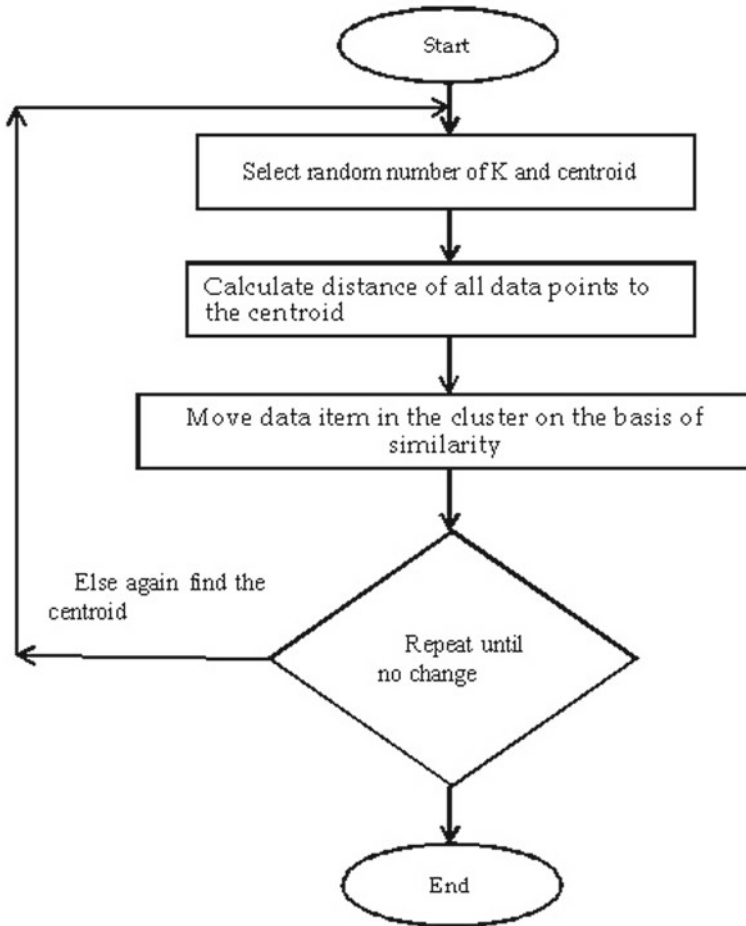


Fig. 1 Flowchart of K-means algorithm

such as (1) the number of K is fixed, (2) noise sensitivity, and (3) it is easy to get lost in the local minima [9]. Because K-means is an NP-hard issue, several evolutionary techniques, such as genetic algorithms, PSO, and machine learning, are utilized to solve it. The steps of the K-means algorithm are listed below, and they demonstrate how the algorithm works:

Method

1. Choose value of K and centroid randomly.
2. Calculate the Euclidean distance from pixel to the centroid for each pixel in an image by using the following relation.

$$D_K(i, j) = \|I_K(i, j) - C_K\| \quad (1)$$

where C_K is the centroid.

1. Assign all pixels to the nearest centroid on the basis of minimum distance value.
2. Recalculate the new position of the centroid by using following relation.

$$C_K = \frac{1}{|S_K|} \sum_{(i,j) \in S_K} I_K(i, j) \quad (2)$$

where $I_K(i, j)$ is the value of the pixel labeled with k th segment and S_K is the set of indices of the pixel belong to class $K \in Y$, where Y is the set of labels of segments.

1. Repeat steps 2–4 until any reassignment occurs.
2. Reshape the cluster pixels into image.

K-means is a cluster analysis approach that is numerical, unsupervised, non-deterministic, iterative, and partition based. It is more common in many research areas due to its simplicity and quick computational capabilities [7].

Arithmetic Mean

The arithmetic mean is simply the average or mean of a series of numbers or records. Different sorts of means are used in mathematics, including arithmetic mean, geometric mean, and harmonic mean. In most K-means clustering algorithms, the objective function is calculated using the arithmetic mean. The arithmetic mean is a decent representation of the average. It has a definite and exact definition. It is easy to calculate and has a fixed value. Arithmetic mean is based on every observation in the data and does not require arranged data. Despite this, the arithmetic mean has some flaws: (1) It is heavily influenced by extreme values. (2) Each value is required for computation, and all objects are counted. (3) Using the graphical approach to find the mean is difficult [10]. Take a set $A = \{a_1, a_2, \dots, a_n\}$ then the arithmetic mean is called \bar{A} and is the mean of the n values

$$a_1, a_2, \dots, a_n.$$

The simplest calculation to find central tendency is the arithmetic mean. The arithmetic mean is calculated by dividing the total number of observations by the sum of their numerical values.

$$AM = \frac{1}{N} \sum_{i=0}^n a_i$$

$$\text{OR } AM = \frac{a_1 + a_2 + \dots + a_n}{n} \quad (3)$$

Harmonic Mean

A harmonic mean is a mean that is determined by multiplying the number of values in a data collection by the sum of their reciprocals ($1/x(i)$). Harmonic is a Pythagorean

term that refers to one of the Pythagorean methods. When calculating the average of ratios or rates, the harmonic mean is widely utilized. It is the best measure for ratios and rates since it equalizes the weights of each data item. Negative or zero values are not allowed in the harmonic mean. The harmonic mean is used in machine learning also. Many researchers have recently proposed various improved algorithms based on K-harmonic mean (KHM) to improve its clustering efficiency. Combining KHM with other heuristic algorithms is one of the most common techniques, as it may take advantage of both the heuristic algorithm's global search capability and KHM's local search capability [11].

Formula for Harmonic Mean

To drive harmonic, mean formula firstly divides the number of values N by the sum of the reciprocal of the values. The harmonic mean (HM) is of N numbers, $n_i \in R, i = 1, 2, \dots, N$ and is defined as follows:

$$HM = N / \left(\sum \frac{1}{X(i)} \right) \quad (4)$$

where HM is the harmonic mean, “ N ” is the total number of observations or values in a data set, and $X(i)$ is the point in a data set [12].

3 Proposed Method

Modified K-means Clustering

The distance function mostly used by the classic K-means method is Euclidean distance. New distances, such as the Manhattan distance and the Minkowski distance, are presented in some journals [13]. The research proposes a new approach for locating the initial centroids of K-means clustering. K-means is a center-based partitioned clustering technique; however, the most notable change in this study is that it calculates the objective function using harmonic means (HM) with arithmetic mean (AM) rather than arithmetic means. The proposed algorithm hybrid K-means (HKM) is not sensitive to the initial centers and has higher clustering performance than K-means with arithmetic mean because of HM's properties of lowest deviation within groups and maximum divergence across groups.

Proposed Algorithm

Let us consider the set of pixels of Image $X = \{x_1, x_2, \dots, x_N\}$ with N number of pixels. A partition $S = \{s^1, s^2, \dots, s^K\}$ where $1 \leq K \leq N$ is the set of the partition of K numbers of segments of image X , such that pixel x_i^j belong to segment s^j .

The modified (hybrid of harmonic and arithmetic means) centroid c_j of s^j is calculated as follows:

$c_j = \sum_{i=1}^{i=M \leq N} \frac{M}{x_i^j} + \sum_{i=1+M}^{i=2M \leq N} \frac{M}{x_i^j} + \dots + \sum_{i=1+IM}^{i=IM \leq N} \frac{M}{x_i^j}$. Here M is the harmonic factor of modified centroid, and N is the number of pixels

$$c_j = \frac{M}{N} \sum_{l=0}^{l=\frac{N}{M}} \sum_{i=(lM+1)M \leq N} \frac{M}{x_i^j} \quad (5)$$

Proposed Algorithm

Initialize the value of K and centroid randomly.
 Calculate the distance on the basis of proposed hybrid K-means (HKM) algorithm using Eq. (5)
 For each pixel in an image from pixel to the centroid.
 Assign all pixels to the nearest centroid on the basis of minimum distance value.
 Recalculate the new position of centroid by using Eq. (5).
 Repeat steps 2–4 until any reassignment occurs.
 Reshape the cluster pixels into image.

4 Result and Discussion

The HKM (proposed algorithm) of image segmentation is used to evaluate different grayscale images. Researchers chose “pepper,” “woman darkhair,” “House,” “Walk-bridge,” “pirate,” and “Cameraman,” as standard test grayscale images from MNIST data sets to evaluate the algorithm’s results, which included efficiency and convergence. The images were captured as TIF files. The initial value of K is 20, and it changes as the evaluation progresses. The cluster size value is taken 5. The algorithm iterates for 100 iterations.

The HKM gives a better result than the existing K-means algorithm. Both the algorithms are compared based on segmentation parameters such as MAE, PSNR, NAE, and RMSE.

For the evaluation of algorithm, the following efficiency factors are used.

$$\text{MAE (Mean Absolute Error)} = \frac{1}{IJ} \sum |X(i, j) - \bar{X}(i, j)|$$

Here $X(i, j)$ and $\bar{X}(i, j)$ are intensity values of pixel of images

$$\text{RMSE (Root Mean Square Error)} = \sqrt{\text{MSE}}$$

$$\text{PSNR (Peak Signal to Noise Ratio)} = \frac{10 \log 255^2}{\text{MSE}}$$

Table 1 Comparison based on PSNR, MAE, RMSE, and NAE values for K-means algorithm

Images	PSNR	MAE	RMSE	NAE
Peppers.tif	66.1632	135.731	9.3288	1.1648
woman_darkhair.tif	65.5856	142.2114	9.6021	1.3229
House.tif	72.2785	97.5622	6.8712	0.7181
Walkbridge.tif	65.4717	124.7144	9.657	1.0968
pirate.tif	65.9018	134.2674	9.4515	1.2069
cameraman.tif	66.3812	146.2115	9.2276	1.2564

Table 2 Comparison based on PSNR, MAE, RMSE, and NAE values for K harmonic means algorithm

Images	PSNR	MAE	RMSE	NAE
Peppers.tif	66.8441	112.4161	9.0165	0.9651
woman_darkhair.tif	67.1557	130.6639	8.8771	1.2162
House.tif	72.2817	83.5487	6.8701	0.6155
Walkbridge.tif	65.7819	118.4179	9.5083	1.0414
pirate.tif	66.6399	129.0974	9.1091	1.1602
cameraman.tif	67.5222	85.3082	8.7159	0.7383

Table 3 Comparison based on PSNR, MAE, RMSE, and NAE values for proposed algorithm hybrid K-means (HKM) algorithm













Images	PSNR	MAE	RMSE	NAE
Peppers.tif	66.9091	111.226	9.2981	0.9606
woman_darkhair.tif	68.1019	130.5797	8.4669	1.217
House.tif	73.5247	90.6801	6.4561	1.1364
Walkbridge.tif	66.2446	118.4135	9.2909	1.0400
pirate.tif	68.2785	129.0949	8.3925	1.1660
cameraman.tif	68.2527	110.9212	8.4033	0.9559

$$(\text{Normalized Absolute Error}) = \frac{\sum |X(i, j) - \bar{A}X(i, j)|}{\sum |X(i, j)|}$$

Here $X(i, j)$, and $\bar{A}(Xi, j)$ are intensity values of pixel of images (Tables 1, 2 and 3).

Above tabulation result clearly shows that the PSNR value of modified algorithm is higher than the existed algorithm, the MAE, NAE, and RMSE are lower than the existed algorithm, and PSNR value is higher than the existed algorithm, which concludes that the modified algorithm is better than the existing algorithm.

Fig. 2 Output images after segmentation by proposed algorithm

Images	Original image	Segmented image
Peppers.tif		
woman_dark		
House.tif		
Walkbridge.tif		
pirate.tif		
cameraman.tif		

Here different standard test images are used for segmentation. Segmentation is found out with the help of proposed (HKM) algorithm. Segmented images are displayed with original images (Fig. 2).

5 Conclusions

A new hybrid K-means (HKM) image segmentation algorithm is proposed in this paper. This algorithm combines the harmonic mean and the arithmetic mean methods. The most popular and straightforward clustering approach is K-means. For pixel classifications, the K-means clustering algorithm is commonly used. Normally, the centroid of the segment is calculated using the arithmetic means, but in this work, the centroid is modified using a hybrid of harmonic and arithmetic means. Several standard test images from the MNIST data set were used to check the proposed algorithm's accuracy. The suggested algorithm (HKM) outperformed existing approaches (K-means and harmonic K-means) in terms of peak signal to noise ratio (PSNR), mean absolute error (MAE), root mean square error (RMSE), and normalized absolute error (NAE). In future, harmonic mean can be used with metaheuristic algorithms for better result.

References

1. Kumari R, Gupta N, Kumar N (2019) Image segmentation using improved genetic algorithm. *Int J Eng Adv Technol (IJEAT)* 9(1):1784–1792
2. Kumari R, Gupta N (2018) Survey on image segmentation techniques using traditional and soft computing techniques. *Int J Comput Sci Eng* 06(05):85–90
3. Kumari R, Gupta N, Kumar N (2020) Cumulative histogram based dynamic particle swarm optimization algorithm for image segmentation. *Indian J Comput Sci Eng* 11(5):557–567
4. Li H, He H, Wen Y (2015) Dynamic particle swarm optimization and K-means clustering algorithm for image segmentation. *Optik* 126(24):4817–4822
5. Panda S (2015) Color image segmentation using K-means clustering and thresholding technique. *Int J ESC* 1132–1136
6. MacQueen J (1967) Some methods for classification and analysis of multivariate observations. In: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol 1, no 14
7. Na S, Xumin L, Yong G (2010) Research on k-means clustering algorithm: an improved k-means clustering algorithm. In: *2010 third international symposium on intelligent information technology and security informatics*. IEEE
8. Dhanachandra N, Manglem K, Chanu YJ (2015) Image segmentation using K-means clustering algorithm and subtractive clustering algorithm. *Procedia Comput Sci* 54:764–771
9. Ding Z, Yu J, Zhang Y (2007) A new improved K-means algorithm with penalized term. In: *2007 IEEE international conference on granular computing (GRC 2007)*. IEEE
10. Medhi J (1992) *Statistical methods: an introductory text*. New Age International
11. Zhou Z, Zhu S, Zhang D (2015) A novel K-harmonic means clustering based on enhanced firefly algorithm. In: *International conference on intelligent science and big data engineering*. Springer, Cham, pp 140–149
12. Hung C-H, Chiou H-M, Yang W-N (2013) Candidate groups search for K-harmonic means data clustering. *Appl Math Model* 37(24):10123–10128
13. Pedrycz W (2005) *Knowledge-based clustering: from data to information granules*. Wiley

A Chatbot for Promoting Cybersecurity Awareness



Yin-Chun Fung and Lap-Kei Lee

Abstract Cybersecurity is one of the hot topics nowadays. However, not many Internet users know the cyber risks around them. To promote cybersecurity awareness, this paper presents a chatbot that is a cybersecurity expert. It aims to let its users learn more about cybersecurity. It relies on Google Dialogflow, which is a natural language understanding platform. Our chatbot contains a knowledge base on cybersecurity knowledge. Users can make queries to the chatbot to learn definitions and concepts of different cybersecurity terms. Our chatbot also provides self-quizzes for users to test their knowledge on different cybersecurity topics. It also provides suggestions to users on cybersecurity issues such as how to identify and handle phishing emails. In a survey of twenty users, the majority of the users agreed that our chatbot is easy to use and can increase their awareness of cybersecurity issues.

Keywords Chatbot · Cybersecurity awareness · Natural language processing

1 Introduction

The term “cybersecurity” started to be famous in 2009 when US President Barack Obama called upon the citizens to recognize the importance of cybersecurity [1]. Cybersecurity awareness is still an important topic around the world, like assessing the awareness of university students [2] and a systematic review of multimedia tools [3]. Yet many cyber users are still unaware of possible cyber risks around them in the cyber world [4]. It is essential to promote awareness of cybersecurity issues in society.

Y.-C. Fung (✉) · L.-K. Lee
School of Science and Technology, Hong Kong Metropolitan University, Ho Man Tin, Kowloon,
Hong Kong SAR, China
e-mail: Chinaycfung@study.ouhk.edu.hk

L.-K. Lee
e-mail: lklee@ouhk.edu.hk

Among the many ways to increase the awareness of cybersecurity, game is one of the popular ways. Pape et al. [5] conceptualized a cybersecurity awareness quiz as a serious game, where the players have to answer questions concerning a social engineering attack, which is one of the common cyber risks. The serious game lets the players think more about security issues in real life. Alqahtani and Kavakli-Thorne [6] developed a game for players to learn cybersecurity using the technology of augmented reality.

Chatbots are artificially intelligent computer software that can chat with humans, which have many applications in the cybersecurity field [7]. One of the applications is the detection of cyber criminals [8], where the chatbot can interact with suspects to profile their interest in online child sexual abuse. The Cyber Helpline¹ in the UK is a chatbot that gathers information about cybercrime incidents and identifies the attack in real time. Artemis is another chatbot example that assists cybersecurity experts, which was developed by Filar et al. [9]. Chatbots can also provide cybersecurity guidelines to users. Gulenko [10] showed a chatbot that can teach the users how to appropriately set the privacy settings on a social media platform and how to set a good password.

Apart from cybercrime detection and analysis, chatbots can be used in education. Nenkov et al. [11] demonstrated how to use a chatbot in Facebook Messenger to let students answer questions of an online test. Lee et al. [12] presented a chatbot for instantly answering students' questions for a university course, and the chatbot supports multiple social platforms commonly used by students, including Telegram, Facebook Messenger and Line. Clarizia et al. [13] also developed a chatbot for supporting students in learning cultural heritage contexts. Some chatbots [14, 15] train one's cybersecurity awareness, but they may not be up to date for fulfilling the cyber environment nowadays.

This paper presents the design of a chatbot to users' awareness of cybersecurity. Our chatbot contains a knowledge base on cybersecurity knowledge. Users can make queries to the chatbot to learn definitions and concepts of different cybersecurity terms. Our chatbot also provides self-quizzes for users to test their knowledge on different cybersecurity topics. It also provides suggestions to users on cybersecurity issues such as how to identify and handle phishing emails. A survey on twenty users from different age groups showed that our chatbot is easy to use and can increase users' awareness of cybersecurity issues.

Organization of the Paper. Section 2 gives the detailed design of our chatbot. Section 3 presents a preliminary evaluation of the chatbot on twenty users. Section 4 concludes the paper and proposes some future work directions.

¹ <https://www.thecyberhelpline.com>.

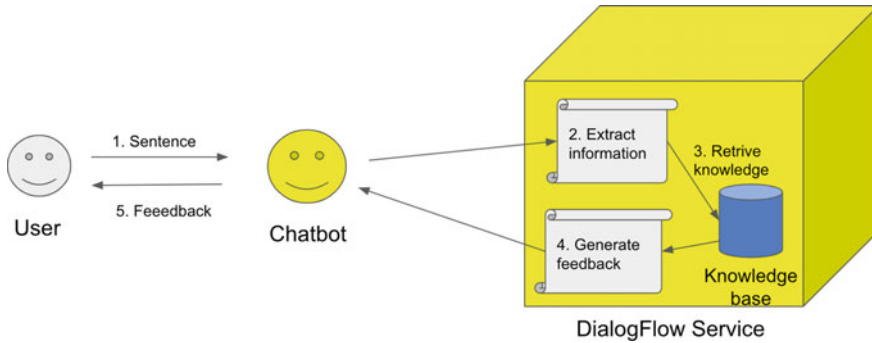


Fig. 1 Architecture of our chatbot

2 Design of the Chatbot

2.1 Architecture

Our chatbot relies on the Google Dialogflow platform,² which is a natural language understanding cloud service. The Google Dialogflow platform contains an inference engine that helps to extract information, including intents and the relevant entities, from messages of the users during a conversation. Figure 1 shows the architecture of the chatbot. When a user message is received, it will be passed to the Dialogflow service to extract the intent and entities of that message. Our chatbot contains a knowledge base on cybersecurity knowledge such that relevant knowledge will be retrieved according to the intent of the message and feedback in English will be generated as the reply to the user.

Dialogflow provides basic features to build a chatbot. First, we need to set up the set of intents and entities such that the chatbot can identify the topics that it needs to handle. Next, the chatbot will be trained using some carefully designed training phrases so that it can correctly identify the intents from different user messages. We can also set up some default responses to different intents. Dialogflow provides a convenient interface for constructing the knowledge base of the chatbot from properly formatted data prepared by us such that the chatbot can retrieve cybersecurity terms and other knowledge and respond to the user queries appropriately.

2.2 Message Handling

Like other chatbots in the market, our chatbot can handle input in different forms.

² <https://cloud.google.com/dialogflow>.

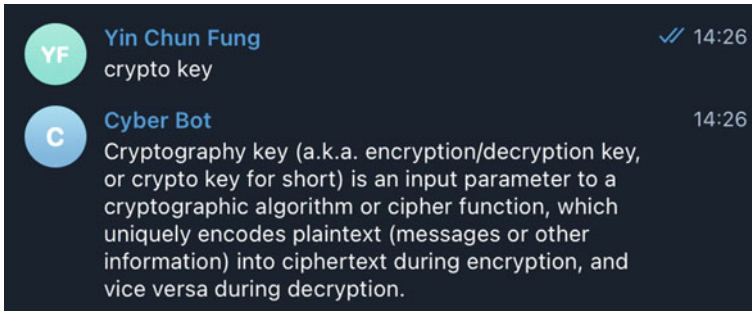


Fig. 2 Example of handling messages in the form of a term



Fig. 3 Example of handling a natural language sentence

Term. Users can input a single term to the chatbot. The chatbot will respond appropriately according to its knowledge base. As shown in Fig. 2, the chatbot obtains the definition of “crypto key” in its knowledge base and response to the user.

Natural Language. Users can input a complete sentence, and the chatbot will analyze the meaning of it. In Fig. 3, with the help of Dialogflow’s inference engine, the chatbot realizes that the user wants to know the definition of malware. It obtains the meaning of the term and then responds to the user.

Command. There are pre-defined commands, which start with a slash (“/”), built in the chatbot. Users can input them, and the chatbot will perform corresponding tasks. For example, in Fig. 4, if the user inputs the command “/quiz,” a self-quiz will be started.

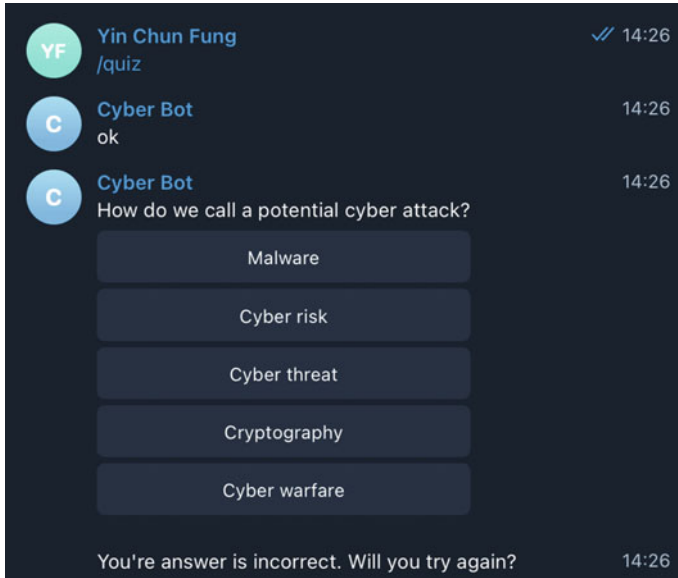


Fig. 4 Example of handling a command

Button. Sometimes the chatbot may require some pre-defined replies from the user. Like that in a quiz shown in Fig. 5, the user should respond in one out of the five choices of a multiple choice question. The buttons are the expected replies from users. Users are not required to input the answer themselves in the chatbox. The buttons also help the chatbot easily trigger feedback from the users. These buttons can also be hyperlinks that link the users to other web pages.

2.3 Supported Features

Term Definition. The knowledge base stores several cybersecurity terms. Users can ask the chatbot about the definition of a term. The chatbot can retrieve the explanation from the knowledge base as shown in Figs. 2 and 3. If the chatbot cannot find a particular term, it will tell users that it does not know the term (Fig. 5).

Self-quiz. User can test their understanding of cybersecurity in a self-test. In a self-test, the chatbot will randomly choose one multiple choice question from the knowledge base to ask the user. If the answer is incorrect, the chatbot will tell the user that the selected answer is incorrect (see Fig. 4 for an example). Users can answer the questions again until the correct answer is chosen. The chatbot will provide a detailed explanation when the answer is correct (Fig. 6).

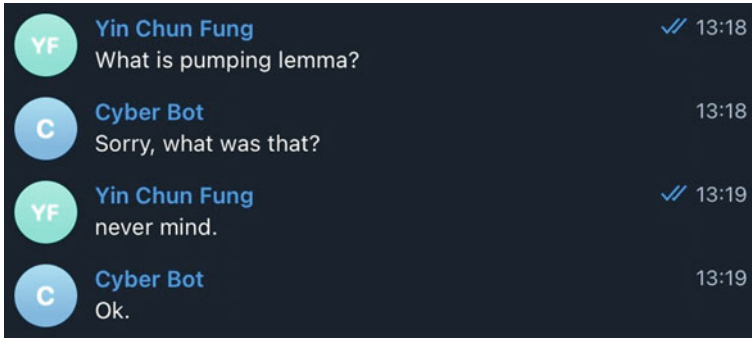


Fig. 5 Example of responses to a term outside the knowledge base

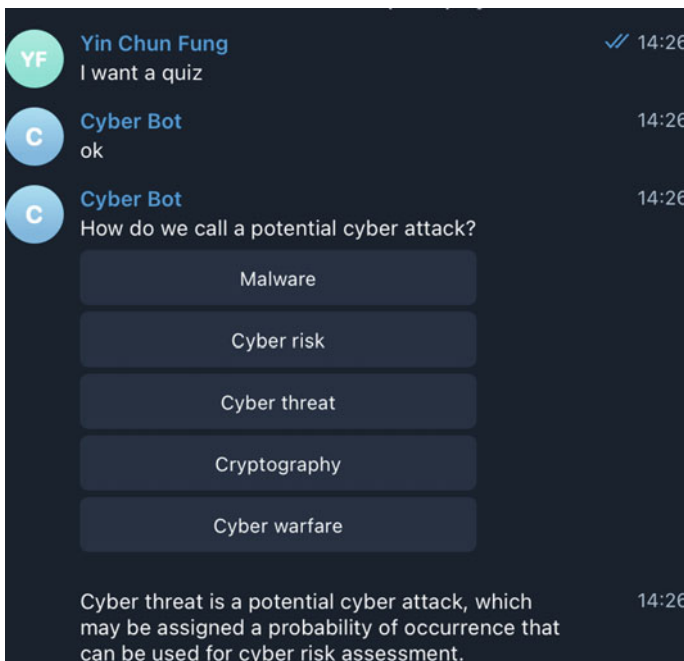


Fig. 6 Example of a self-quiz question

Workflow Structure. The chatbot provides some workflow to help the user with cybersecurity issues. One of the workflows is to help users determine whether an email is a phish. It will ask the users questions and follow the decision flow in Fig. 7. According to the responses from users, the chatbot can tell what the user should do regarding the email (Fig. 8).

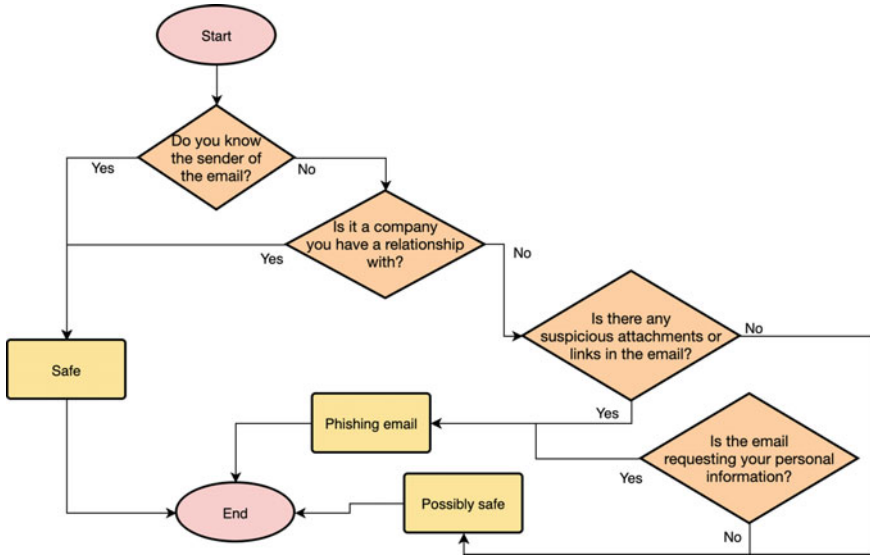


Fig. 7 Decision flow of the identification of phishing emails

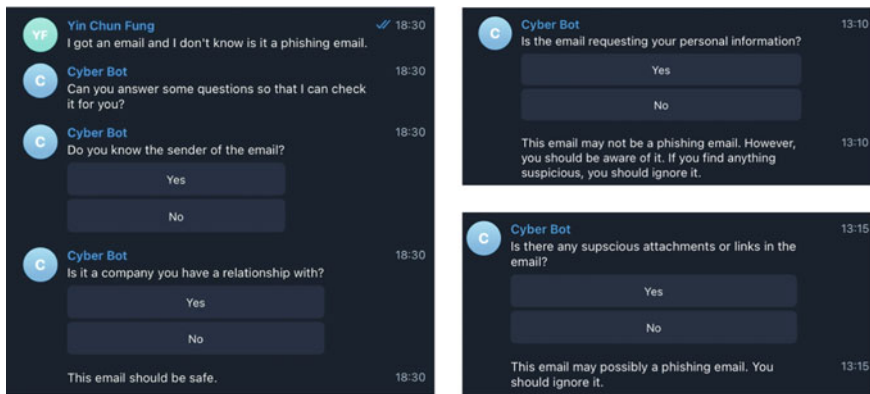


Fig. 8 Example on how to identify a phishing email

3 Preliminary Evaluation

We invited 20 participants from the Internet to chat with our chatbot and complete a survey. They are of different industries and have different backgrounds in cybersecurity. The survey consists of seven questions on a 5-point Likert scale (1: disagree, 2: partially disagree, 3: neutral, 4: partially agree, 5: agree) and one open-ended question to give some comment in the text about the chatbot. Table 1 shows the survey results.

Table 1 Survey result

Item	1	2	3	4	5
The chatbot is easy to use (%)	0	0	25	70	5
The chatbot can answer questions correctly (%)	0	0	30	60	10
The content given by the chatbot is easy to understand (%)	0	15	30	50	5
The chatbot is a quick tool for checking cybersecurity terms (%)	0	0	15	80	5
The quiz by the chatbot is useful (%)	0	10	65	15	10
The answer to the quiz by the chatbot is accurate (%)	0	0	0	95	5
The chatbot can make me more aware of cybersecurity (%)	0	0	25	60	15

In the evaluation, Question 1 reflects the ease of use of the chatbot. 75% of users agree that the chatbot is easy to use. 25% of users think it is neutral. They suggest that the list of commands and their usage is not clear.

Questions 2–4 correspond to the question answering of cybersecurity knowledge. More than half of the respondents think that the chatbot gives a satisfactory response. However, some users report that they cannot get the definition of networking terms like TCP/IP which they think is related to cybersecurity and the chatbot should be able to give a brief explanation on it.

Questions 5–6 concern the self-quiz given by the chatbot. Users think it will be great to have regular quizzes with more questions. A scoreboard can be used to record users' scores to make the quiz more competitive.

Question 7 asks the user if the chatbot makes them more aware of cybersecurity issue. 75% of them agree that it is true that our chatbot increases their awareness of cybersecurity.

Some users also think that the chatbot can provide more functionality like suggesting the strength of a password and remind them to change passwords at a period regularly. They also suggest that the bot can give some cybersecurity reading daily to keep their awareness.

4 Conclusion and Future Work

Cybersecurity awareness is an important topic in nowadays society. This paper presented a chatbot to promote cybersecurity awareness. It contains a knowledge base with cybersecurity terms. It can explain terms to users. Users can take self-quizzes to test their understanding of cybersecurity knowledge and revise the knowledge they learned from the chatbot. It also provides workflows to assist the user in some cybersecurity issues such as determining a phishing email. In the evaluation, majority of the respondent agrees that the chatbot makes them more aware of cybersecurity.

There are still a lot of improvements to the chatbot. The chatbot should provide more functionality and provide more workflows to assist users in multiple aspects.

The chatbot can be implemented into personal assistant apps to remain the users about potential threats of cybersecurity. The chatbot should also fill up with terms that are less relating to cybersecurity but are important when learning terms in cybersecurity into its knowledge base. These will be our future works.

References

1. Sanger DE, Markoff J (2009) Obama outlines coordinated cyber-security plan. *The New York Times*, p 29
2. Alharbi T, Tassaddiq A (2021) Assessment of cybersecurity awareness among students of Majmaah University. *Big Data Cogn Comput* 5(2):23
3. Zhang-Kennedy L, Chiasson S (2021) A systematic review of multimedia tools for cybersecurity awareness and education. *ACM Comput Surv* 54(1):1–39
4. Nagyfejeo E, von Solms B (2020) Why do national cybersecurity awareness programmes often fail?
5. Pape S, Goeke L, Quintanar A, Beckers K (2020) Conceptualization of a cybersecurity awareness quiz. In: *International workshop on model-driven simulation and training environments for cybersecurity*, Sept 2020. Springer, Cham, pp 61–76
6. Alqahtani H, Kavakli-Thorne M (2020) Design and evaluation of an augmented reality game for cybersecurity awareness (CybAR). *Information* 11(2):121
7. Mármol FG, Pérez MG, Pérez GM (2016) I don't trust ICT: research challenges in cyber security. In: *IFIP international conference on trust management*, July 2016. Springer, Cham, pp 129–136
8. Rodríguez JI, Durán SR, Díaz-López D, Pastor-Galindo J, Mármol FG (2020) C3-sex: a conversational agent to detect online sex offenders. *Electronics* 9:1779
9. Filar B, Seymour R, Park M (2017) Ask me anything: a conversational interface to augment information security workers. In: *SOUPS 3rd workshop on security information workers (WSIW 2017)*, July 2017
10. Gulenko I (2014) Chatbot for IT security training: using motivational interviewing to improve security behaviour. In: *AIST (supplement)*, pp 7–16
11. Nenkov N, Dimitrov G, Dyachenko Y, Koeva K (2016) Artificial intelligence technologies for personnel learning management systems. In: *2016 IEEE 8th international conference on intelligent systems (IS)*, Sept 2016. IEEE, pp 189–195
12. Lee LK, Fung YC, Pun YW, Wong KK, Yu MTY, Wu NI (2020) Using a multiplatform chatbot as an online tutor in a university course. In: *2020 international symposium on educational technology (ISET)*. IEEE, pp 53–56
13. Clarizia F, Colace F, Lombardi M, Santaniello D (2020) A chatbot for supporting users in cultural heritage contexts
14. Al Sabbagh B, Ameen M, Wätterstam T, Kowalski S (2012) A prototype for HI²Ping information security culture and awareness training. In: *2012 international conference on E-learning and E-technologies in education (ICEEE)*, Sept 2012. IEEE, pp 32–36
15. Kowalski S, Pavlovskaa K, Goldstein M (2013) Two case studies in using chatbots for security training. In: *Information assurance and security education and training*. Springer, Berlin, Heidelberg, pp 265–272

An Advanced Irrigation System Using Cloud-Based IoT Platform ThingSpeak



Salman Ashraf and A. Chowdhury

Abstract The conventional irrigation systems are manual and thus require human effort and interruption to water the crops, and in such a scenario, water wastage is evident. This work was designed to address these two problems associated with conventional irrigation systems, i.e. manual operation and water wastage. The system designed has been made smart and thus has automatic decision-making ability to reduce human effort and interruption. The designed system has made use of sensors like soil moisture sensor, temperature and humidity sensor, and rain sensor and thus can calculate the moisture content of the soil, read surrounding temperature and humidity, and sense rainfall. It has the feature of making an application programming interface (API) weather call to extract information about rainfall from the OpenWeather webpage and finally makes a decision comparing all collected data and threshold data already set by the user. ThingSpeak, a cloud-based Internet of Things (IoT) platform, has been used for storing the data read by various sensors in the form of graphs for better visualization and future reference. A (Global System for Mobile Communication-Global Positioning System) GSM-GPS module is also taken into work for establishing Internet connection and determining the system location for precise weather data. The system was tested for different threshold values of soil moisture and temperature reading, and based on the comparison with real-time sensor values, it successfully turned ON/OFF the motor and thus found to work fine as desired. The weather data fetched by the system also found to match with the real-world weather conditions.

Keywords Smart irrigation · Soil moisture sensor · Rain sensor · Application programming interface (API) · OpenWeather · Internet of Things (IoT) · ThingSpeak

S. Ashraf (✉) · A. Chowdhury
Electronics and Communication Engineering Department, NIT Agartala, Agartala, India
e-mail: salmanashraf.16@gmail.com

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
D. P. Agrawal et al. (eds.), *Cyber Security, Privacy and Networking*, Lecture Notes
in Networks and Systems 370, https://doi.org/10.1007/978-981-16-8664-1_34

389

1 Introduction

1.1 Motivation

India is not a country of abundant water resources, and many of its states are predicted to face severe drought soon. This is due to the wastage of water owing to poor management. Now, as we know agriculture is the primary consumer of water and if we can design and implement systems that can reduce the wastage in agricultural fields, then we can save a lot of water every year. To achieve this efficiency, manual systems of irrigation are not good enough and need a technology-aided system which may be automatic or semi-automatic and hence comes into picture the technology-aided system which is better known as smart irrigation systems.

1.2 Literature Review

Pednekar et al. [1] have proposed a system using a soil moisture sensor, microcontroller, and Zigbee. Zigbee is used for wireless communication between sender and receiver. In addition to this, a GSM module has also been added for alerting the user regarding the moisture content of the soil.

Thamaraimanalan et al. [2] have discussed designing an IoT-based system using sensors and NodeMCU. Sensors like a soil moisture sensor, a temperature and humidity sensor, and an ultrasonic sensor have been used for the application. An android application has been developed and taken into use for remote monitoring and controlling the system.

Bafna et al. [3] have developed a system using Arduino Uno, NodeMCU, and soil moisture sensor. For storing data, the Firebase application has been taken into use. They have also developed an android application, wherein location-based weather prediction is done using mobile GPS and weather API, and based on this prediction and sensor data, users can decide whether to turn ON the motor or not.

A system using cloud-based application ThingSpeak, a light sensor, and a soil moisture sensor that have been interfaced with Arduino Uno and Arduino Wi-Fi shield has been proposed and developed by Al-Omary et al. [4]. The ThingSpeak channel has been used to store the threshold values of moisture content of soil and the amount of light required for plants to enable the system to make decisions based on those threshold values and sensor readings.

Garg et al. [5] have reviewed some of the soil moisture sensors already available and discussed their specifications, properties, applicability, advantages, and disadvantages, hence making it easier to select which type of sensor will be suitable for us.

Arduino-based smart irrigation system using GSM module and sun tracking solar system has been explored by Karmokar et al. [6]. The solar tracking feature is the

main focus of this project. This feature helps to consume the maximum energy from the sun, while the sun changes its azimuth over the day.

Hatanaka et al. [7] have talked about the use of a tensiometer, which works on the principle of measuring the pressure force required to extract moisture from the porous cap presented in the soil, as a potential way of measuring soil moisture. They designed the system as a client–server system, where the client is the user who uses the interface developed to fetch data from the server. The server system consists of the grove moisture sensor along with Arduino Uno and Ethernet shield.

Jain and Kumar [8] have developed a smart irrigation system using sensors like soil moisture sensor, water-level sensor, temperature sensor, humidity sensor, and an ARM7 processor. A water-level sensor is used to keep track of the water present in a water tank.

A system was developed by Athani et al. [9] where a soil moisture sensor is connected to an Arduino which is, in turn, interfaced with an android application using a Wi-Fi shield. Soil moisture is continuously monitored by the sensor, and the output values are stored in a database. The output values are fetched from the database and displayed in the android application.

Divya Dhatri et al. [10] have designed a low-cost Arduino-based irrigation system using a soil moisture sensor. An LCD module is used to display the relevant information and status of the motor.

Prasojo et al. [11] have designed a smart irrigation system using a basic sensor like soil moisture sensor and Arduino Uno along with other modules like relay module, LCD module, and solenoid valve motor.

A detailed discussion on soil water content and threshold limits for irrigation management has been done by Datta et al. [12]. The terms related to soil moisture condition like volumetric water content, soil matric potential, saturation, and field capacity have been introduced and discussed in detail.

Patel et al. [13] have discussed an Arduino-based irrigation system that consists of sensors connected to it and an ESP8266 module for Internet connectivity to send data to the ThingSpeak channel. It has a wireless sensor network for real-time sensing of irrigation systems. This system avoids wastage of water by automatically switching ON the motor when the moisture level in the soil reaches below a threshold value. All current statuses of the system will be displayed in the user's android application.

A discussion on implementing an irrigation system based on the Internet of Things (IoT) using ESP8266 NodeMCU and temperature and humidity sensor has been done by Anitha et al. [14]. The ThingSpeak server is taken into use to keep track of the moisture level in the soil and to store the data in the IoT cloud. Also, rain alarm and soil moisture detector circuits are used to build the smart irrigation system.

Naeem et al. [15] have discussed developing an irrigation system by integrating a real-time monitoring system having remote controllability and cloud computation of stored data. In addition, a mobile application has also been developed for a better user experience. They have designed the system using various sensors and water-level detector in addition to NodeMCU and Arduino development board.

An irrigation system was developed by Karpagam et al. [16] using Arduino UNO, soil moisture sensor, water-level sensor, temperature and humidity sensor, and GSM

module. With the help of the GSM module, the information regarding the ON and OFF states of the water pump is sent to the user.

Sen et al. [17] have utilized Arduino UNO and ESP8266 Wi-Fi module to design their system. Arduino collects all sensor data and sent them to the ThingSpeak channel using the ESP module.

Smart irrigation is of priority to many researchers nowadays, and it is evident from the fact that a lot of work have been done concerning it. Research works on smart irrigation have been studied, and based on the shortcomings of the available research work, this work has been executed. The research works discussed above have some limitations like without a rain sensor the system would not be able to sense if it is raining and thus turn OFF the motor in case it is turned ON at that instant. Also, rain prediction using weather API and location extraction using GPS module will help the existing system extensively in making accurate decision. These two features have been added in this work to make the system truly smart decision-making device.

2 System Design and Implementation

The architecture of the designed system prototype is shown in Fig. 1. It shows the microcontroller in the centre as the decision-making hub, and all the sensors and modules are connected to it. The phone block shown in the block diagram is not connected directly to the microcontroller but is shown as a symbol to signify that it is a part of the working system. The phone is used in the system as an additional component to alert the user by sending SMS.

Figure 2 shows the implemented hardware. It shows the actual sensors and modules used in this work.

Figure 3 shows the flow chart of the system. First, the system is turned ON by supplying power to the Arduino and the GSM-GPS module. As soon as the system is turned ON, the soil moisture sensor reads the moisture content of the soil, and the temperature and humidity sensor reads the temperature and humidity of the place surrounding that system. These collected data are then sent to a ThingSpeak channel where we can store them.

The GPS integrated along with the GSM module determines the geographical coordinates of the location where the system is placed in the form of latitude and longitude. An API weather call is made in the OpenWeather website by making use of the determined coordinates which fetches weather data of the given location in JSON format and contains information related to rain, wind speed, wind direction, clouds, etc. The rain sensor module used in the system keeps checking the wetness of the board for determining if it is raining. Now, all the data that are collected by the sensors will be compared against their respective threshold limits already set by the user. If the result of comparison meets the set threshold limits, then the input signal of the relay is set high which would turn on the motor till threshold moisture condition is not reached.

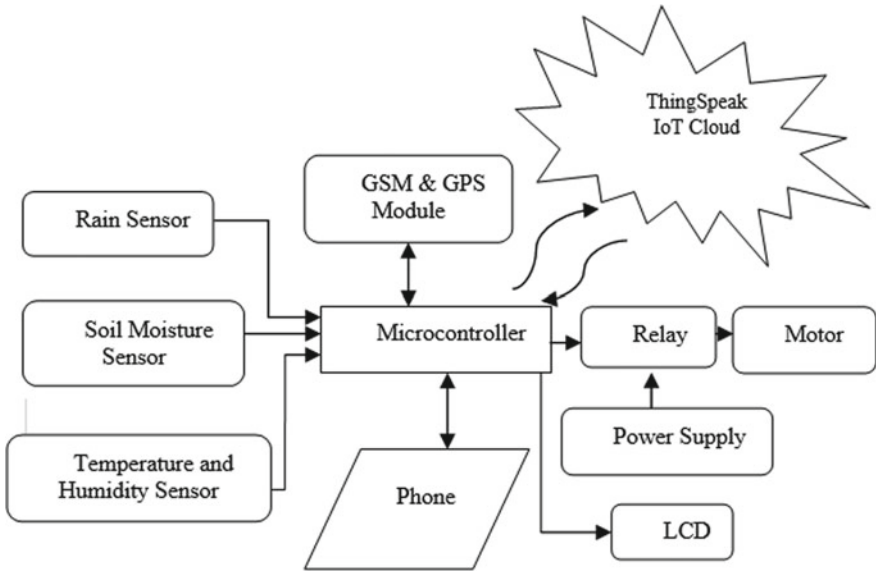


Fig. 1 System architecture

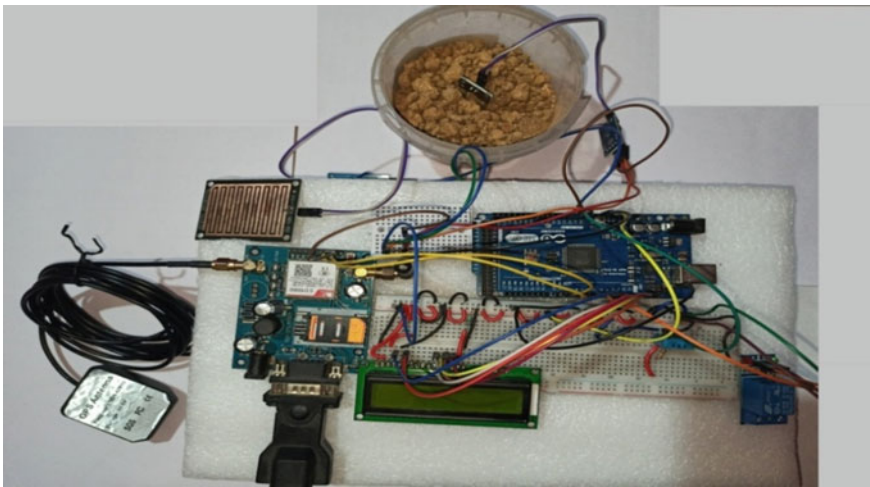


Fig. 2 Implemented hardware

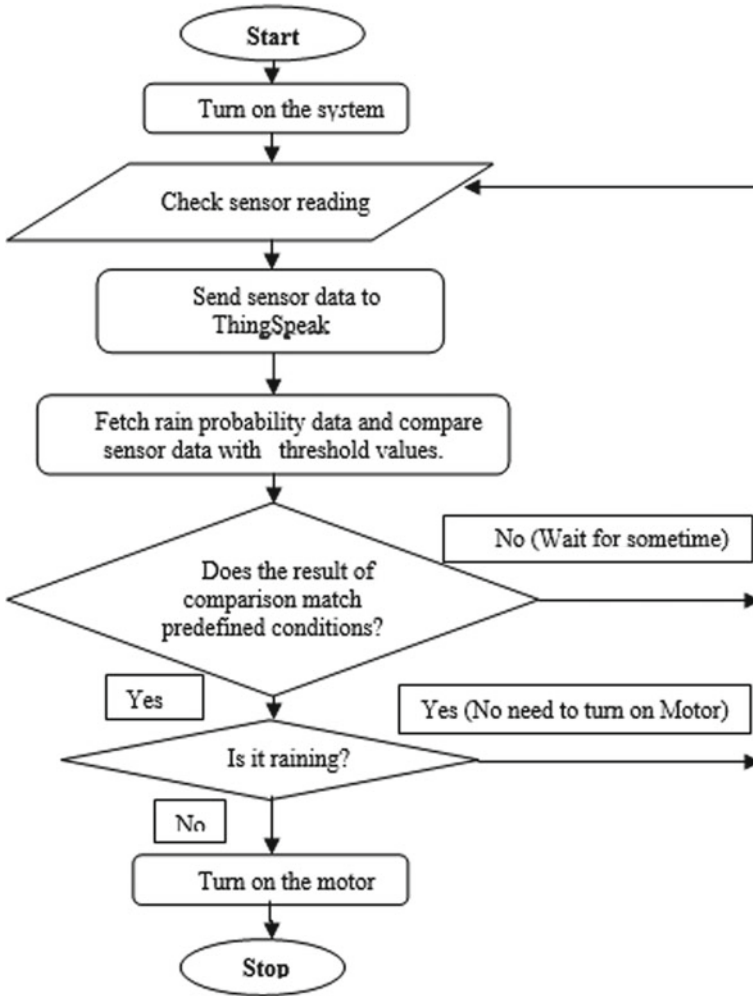


Fig. 3 Flow chart

3 Results and Discussion

ThingSpeak, a cloud-based IoT platform from Mathworks has been used for storing the sensor data sent by the Arduino microcontroller. These data are plotted in ThingSpeak in the form of graphs which give a better visualization of the data. Some of the screenshots of the plotted data are shown in Fig. 4a–e. Also, ThingSpeak has the feature of remotely controlling the system based on the analysis of the stored data.

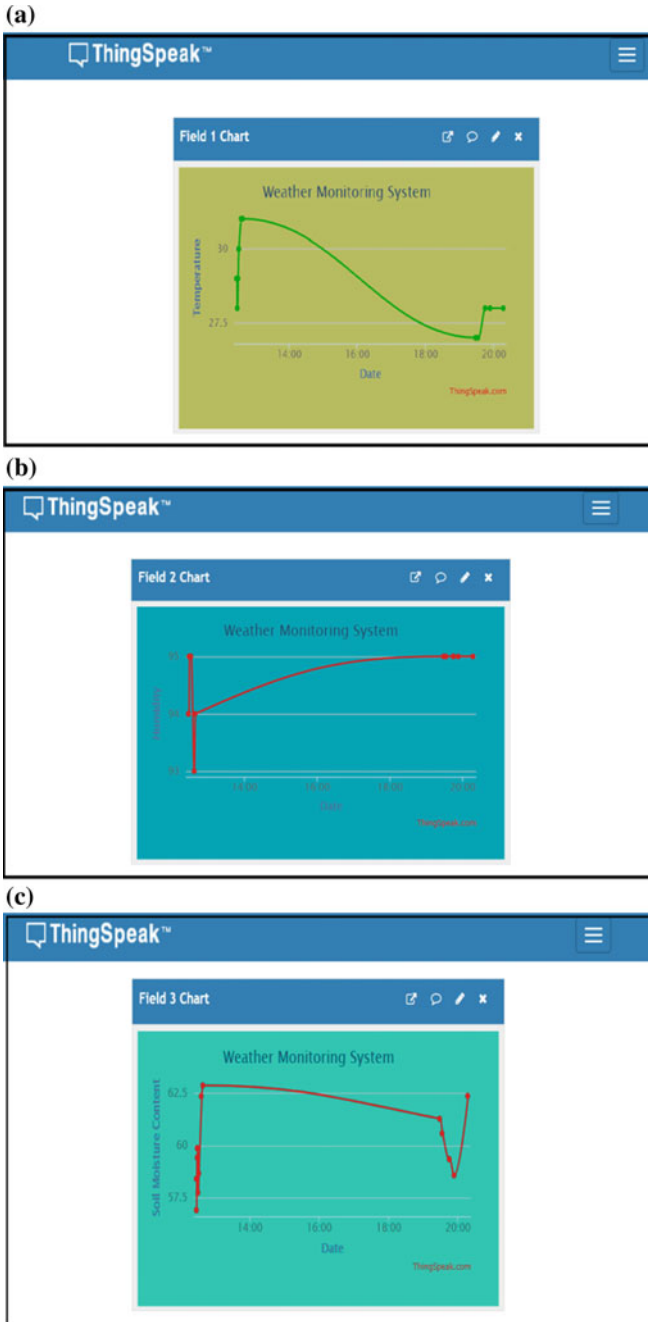


Fig. 4 a Temperature field. b Humidity fields. c Soil moisture content field. d Temperature threshold field. e Soil moisture threshold fields

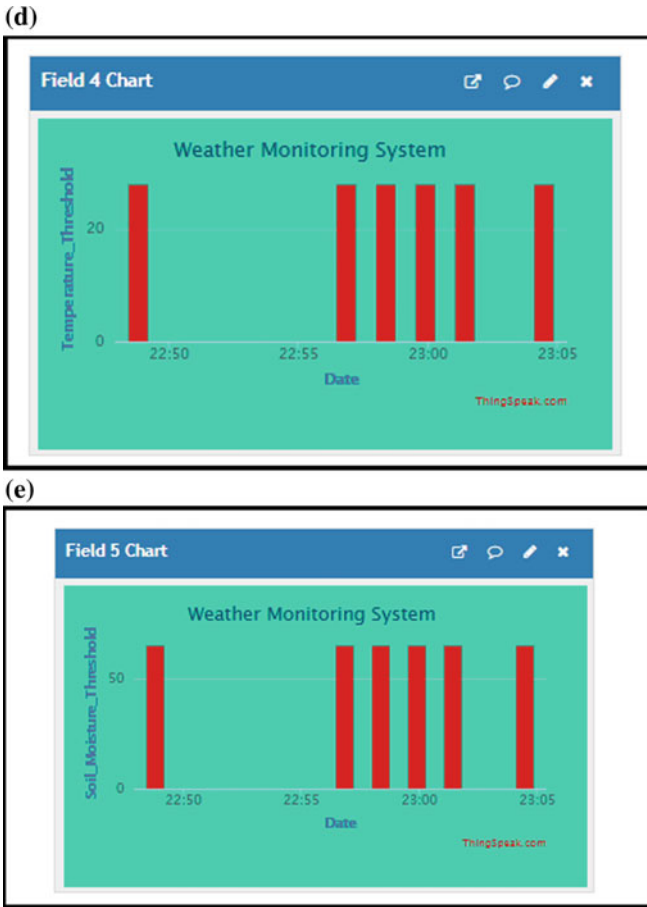


Fig. 4 (continued)

Figure 4a shows the plot of temperature reading from the DHT11 sensor versus time. The graph shows the values of temperature recorded along the y-axis at different instants of time. Since there are slight variations in temperatures recorded, the graph seems to have abrupt changes in readings.

Figure 4b shows the reading of humidity recorded by the sensor at different instants of time. The maximum and minimum humidity recorded were 95% and 93%, respectively. As the variation in humidity is very less, hence the graph is having abrupt changes, i.e. spikes.

Figure 4c shows the graph of soil moisture content of the soil of the location where the device was placed. The system was tested for its performance at different levels of soil moisture content, and hence, there are variations in reading as seen in the graph.

Figure 4d, e represents the temperature threshold, soil moisture threshold and are constant that depend on the type of soil and crops grown. These values are stored in the ThingSpeak channel in the form of bar graphs and not as continuous line graphs at different instants of time and hence seem to have some values missing.

4 Conclusion

A prototype of an IoT-based advanced irrigation system has been designed and implemented using different sensors and modules. This system was developed to provide a cost-effective, reliable, and user-friendly system for the mass to achieve the goal of smart irrigation to a great extent. In this work, we have utilized the cloud-based application “ThingSpeak” to store, visualize, and analyse the data captured by various sensors. All the data were sent continuously to the ThingSpeak channel over the Internet. GSM module was used for two specific reasons: first to solve the problem of Internet connectivity in any remote location and second to send SMS alerts to the user. The system thus developed can take readings from the sensors interfaced with Arduino, sent them to the cloud for storage and analysis, fetch rain probability data, sense rainfall, and finally combine all these data to make a decision.

The smart irrigation system discussed in this work is found to perform better when compared to similar types of existing systems in terms of rainfall sensing and prediction due to the use of a rain sensor and API weather call.

References

1. Pednekar S, Lohani RB, Gawde G (2011) Soil moisture sense to trigger irrigation. In: International conference on advancements in information technology. IACSIT Press, Singapore, pp 72–76
2. Thamaraimanalan T, Vivekk SP, Satheeshkumar G, Saravanan P (2018) Smart garden monitoring system using IOT. *Asian J Appl Sci Technol (AJAST)* 2(2):186–192
3. Bafna A, Jain A, Shah N, Parekh R (2018) IoT based irrigation using Arduino and android on the basis of weather prediction. *Int Res J Eng Technol (IRJET)* 5(4):433–437
4. Al-Omary A, AlSabbagh HM, Al-Rizzo H (2018) Cloud based IoT for smart garden watering system using Arduino Uno. In: Smart cities symposium 2018. Bahrain, pp 1–6
5. Garg A, Munoth P, Goyal R (2016) Application of soil moisture sensors in agriculture: a review. In: Proceedings of international conference on hydraulics, water resources and coastal engineering. Pune, pp 1662–1672
6. Karmokar C, Hasan J, Khan SA, Alam MII (2018) Arduino UNO based smart irrigation system using GSM module, soil moisture sensor, sun tracking system and inverter. In: 2nd international conference on innovations in science, engineering and technology (ICISSET). IEEE, Chittagong, Bangladesh, pp 98–101
7. Hatanaka D, Ahrary A, Ludena D (2015) Research on soil moisture measurement using moisture sensor. In: 4th international congress on advanced applied informatics. IEEE, Okayama, Japan, pp 663–668
8. Jain A, Kumar A (2020) Smart agriculture monitoring system using IoT. *Int J Res Appl Sci Eng Technol (IJRASET)* 8(6):366–372

9. Athani S, Tejeshwar CH, Patil MM, Patil P, Kulkarni R (2017) Soil moisture monitoring using IoT enabled Arduino sensors with neural networks for improving soil management for farmers and predict seasonal rainfall for planning future harvest in North Karnataka—India. In: International conference on I-SMAC (IoT in social, mobile, analytics, and cloud). IEEE, Palladam, India, pp 43–48
10. Divya Dhatri PVS, Pachiyannan M, Jyothi Swaroopa Rani K, Pravallika G (2019) A low-cost Arduino based automatic irrigation system using soil moisture sensor: design and analysis. In: International conference on signal processing and communication. IEEE, Coimbatore, pp 104–108
11. Prasajo I, Maselena A, Tanane O, Shahu N (2020) Design of automatic watering system based on Arduino. *J Robot Control (JRC)* 1(2):55–58
12. Datta S, Taghvaeian S, Stivers J. Understanding soil water content and thresholds for irrigation management
13. Patel J, Patel E, Pati P (2019) Sensor and cloud based smart irrigation system with Arduino: a technical review. *Int J Eng Appl Sci Technol* 3:25–29
14. Anitha A, Sampath N, Jerlin MA (2020) Smart irrigation system using Internet of Things. In: International conference on emerging trends in information technology and engineering (IC-ETITE). IEEE, Vellore, pp 1–7
15. Naeem MRH, Gawhar S, Adib MBH, Sakib SA, Ahmed A, Chisty NA (2021) An IoT based smart irrigation system. In: 2nd international conference on robotics, electrical and signal processing techniques (ICREST). IEEE, Dhaka, Bangladesh, pp 243–247
16. Karpagam J, Merlin II, Bavithra P, Kousalya J (2020) Smart irrigation system using IoT. In: 6th international conference on advanced computing and communication systems (ICACCS). IEEE, Coimbatore, pp 1292–1295
17. Sen, D, Dey M, Kumar S, Boopathi CS (2020) Smart irrigation using IoT. *Int J Adv Sci Technol* 29(4s):3080–3090

Index

A

Aayush Agarwal, 83
A. Chowdhury, 389
Aditya Kulkarni, 49
Akash Bagade, 49
Akshat Gaurav, 175
Aleksandar Stjepanovic, 309
Anupama Mishra, 165, 339
Apoorva Jain, 61
A. Saranya, 95
Ayush Hans, 155

B

B.B. Gupta, 175
Bina Kotiyal, 219

D

Dalibor Dobrilovic, 309
Devangi Purkayastha, 13
Dr. Deepak Sharma, 105
Dr. Heman Pathak, 357
Dr. Swati Shinde, 25, 37
Dr. S. Ramamoorthy, 321
D. Malathi, 13
Dragan Perakovic, 309

E

Ekansh, 155

F

Faisal Anwer, 185

G

Garima Bajaj, 339
Gauri Pawar, 25
Goran Jausevac, 309
Gordana Jotanovic, 309
Govind P. Gupta, 347

H

Harnain Kour, 117
Heman Pathak, 219

J

Jagadish S. Kallimani, 197
Jingjing Wang, 287
J. Sangeetha, 259, 273
J. Y. Srikrishna, 259

K

K. Janani, 321
K. Muthumanickam, 231
Kunal Ravindra Mohadikar, 155
Kwok Tai Chui, 175

L

Lap-Kei Lee, 287, 297, 379

M

Manoj Kumar Gupta, 117, 129
M. H. Chaitra, 247
M. Vijayakumar, 141

N

Narendra Singh Yadav, 207
 Neena Gupta, 165, 369
 Neha Sharma, 207
 Nga-In Wu, 287, 297
 Nisheeth Srivastava, 61

P

P. N. K. Varalakshmi, 197
 P. Pandiaraja, 231
 Prachi Nangare, 49
 Pragati Janjal, 25
 Prajakta Shinde, 49
 Prasad Borle, 37
 Prof. Grishma Sharma, 105

R

Rachna Sharma, 71
 Raza Imam, 185
 Ringo Pok-Man Leung, 297
 R. Naresh, 95
 Roopa Kumari, 369
 Rutuja Rashinkar, 25

S

Salman Ashraf, 389
 Sanjay Kumar, 347
 Santosh K. Simmarwar, 347
 Santwana Gudadhe, 49

Saurabh Kulkarni, 37
 Saurabh Tailwal, 339
 Shivangi Shah, 71
 Siddhant Bhatnagar, 71
 Subhashini Gupta, 105
 Suyash Khachane, 37
 S. Vagdevi, 247
 Syeda Sabah Sultana, 273
 Swapnil Rokade, 25

T

Tanvi Mahajan, 37
 T.S. Shiny Angel, 141

U

Umesh Kumar, 1, 83

V

V. Ch. Venkaiah, 1, 83

Y

Yashi Chaudhary, 357
 Yin-Chun Fung, 379

Z

Zahid Maqsood, 129