Lecture Notes in Networks and Systems 473

Deepak Gupta · Ashish Khanna · Siddhartha Bhattacharyya · Aboul Ella Hassanien · Sameer Anand · Ajay Jaiswal *Editors*

International Conference on Innovative Computing and Communications

Proceedings of ICICC 2022, Volume 1

Springer

Lecture Notes in Networks and Systems

Volume 473

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).

Deepak Gupta · Ashish Khanna ·
Siddhartha Bhattacharyya · Aboul Ella Hassanien ·
Sameer Anand · Ajay Jaiswal
Editors

International Conference on Innovative Computing and Communications

Proceedings of ICICC 2022, Volume 1

Springer

Editors
Deepak Gupta
Department of Computer Science Engineering
Maharaja Agrasen Institute of Technology
Rohini, Delhi, India

Ashish Khanna
Department of Computer Science Engineering
Maharaja Agrasen Institute of Technology
Rohini, Delhi, India

Siddhartha Bhattacharyya
Rajnagar Mahavidyalaya
Birbhum, West Bengal, India

Aboul Ella Hassanien
Department of Information Technology
Cairo University
Giza, Egypt

Sameer Anand
Department of Computer Science
Shaheed Sukhdev College of Business Studies
University of Delhi
New Delhi, Delhi, India

Ajay Jaiswal
Department of Computer Science
Shaheed Sukhdev College of Business Studies
University of Delhi
New Delhi, Delhi, India

ISSN 2367-3370 ISSN 2367-3389 (electronic)
Lecture Notes in Networks and Systems
ISBN 978-981-19-2820-8 ISBN 978-981-19-2821-5 (eBook)
https://doi.org/10.1007/978-981-19-2821-5

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

Dr. Deepak Gupta would like to dedicate this book to his father Sh. R. K. Gupta, his mother Smt. Geeta Gupta for their constant encouragement, his family members including his wife, brothers, sisters, kids, and his students close to his heart.

Dr. Ashish Khanna would like to dedicate this book to his mentors Dr. A. K. Singh and Dr. Abhishek Swaroop for their constant encouragement and guidance and his family members including his mother, wife, and kids. He would also like to dedicate this work to his (Late) father Sh. R. C. Khanna with folded hands for his constant blessings.

Prof. (Dr.) Siddhartha Bhattacharyya would like to dedicate this book to Late Kalipada Mukherjee and Late Kamol Prova Mukherjee.

Prof. (Dr.) Aboul Ella Hassanien would like to dedicate this book to his wife Nazaha Hassan.

Dr. Sameer Anand would like to dedicate this book to his Dada Prof. D. C. Choudhary, his beloved wife Shivanee and his son Shashwat.

Dr. Ajay Jaiswal would like to dedicate this book to his father Late Prof. U. C. Jaiswal, his mother Brajesh Jaiswal, his beloved wife Anjali, his daughter Prachii, and his son Sakshaum.

ICICC-2022 Steering Committee Members

Patrons

Dr. Poonam Verma, Principal, SSCBS, University of Delhi
Prof. Dr. Pradip Kumar Jain, Director, National Institute of Technology Patna, India

General Chairs

Prof. Dr. Siddhartha Bhattacharyya, Christ University, Bangalore
Dr. Prabhat Kumar, National Institute of Technology Patna, India

Honorary Chairs

Prof. Dr. Janusz Kacprzyk, FIEEE, Polish Academy of Sciences, Poland
Prof. Dr. Vaclav Snasel, Rector, VSB-Technical University of Ostrava, Czech Republic

Conference Chairs

Prof. Dr. Aboul Ella Hassanien, Cairo University, Egypt
Prof. Dr. Joel J. P. C. Rodrigues, National Institute of Telecommunications (Inatel), Brazil
Prof. Dr. R. K. Agrawal, Jawaharlal Nehru University, Delhi

Technical Program Chairs

Prof. Dr. Victor Hugo C. de Albuquerque, Universidade de Fortaleza, Brazil
Prof. Dr. A. K. Singh, National Institute of Technology, Kurukshetra
Prof. Dr. Anil K. Ahlawat, KIET Group of Institutes, Ghaziabad

Editorial Chairs

Prof. Dr. Abhishek Swaroop, Bhagwan Parshuram Institute of Technology, Delhi
Dr. Arun Sharma, Indira Gandhi Delhi Technical University for Women, Delhi

Organizing Secretaries

Dr. Ajay Jaiswal, SSCBS, University of Delhi
Dr. Sameer Anand, SSCBS, University of Delhi
Dr. Ashish Khanna, Maharaja Agrasen Institute of Technology (GGSIPU), New Delhi
Dr. Deepak Gupta, Maharaja Agrasen Institute of Technology (GGSIPU), New Delhi
Dr. Gulshan Shrivastava, National Institute of Technology Patna, India

Publication Chair

Dr. Vicente García Díaz, University of Oviedo, Spain

Publicity Chairs

Dr. M. Tanveer, Indian Institute of Technology, Indore, India
Dr. Jafar A. Alzubi, Al-Balqa Applied University, Salt, Jordan
Dr. Hamid Reza Boveiri, Sama College, IAU, Shoushtar Branch, Shoushtar, Iran

Co-convener

Mr. Moolchand Sharma, Maharaja Agrasen Institute of Technology, India

Organizing Chairs

Dr. Kumar Bijoy, SSCBS, University of Delhi
Dr. Rishi Ranjan Sahay, SSCBS, University of Delhi
Dr. Amrina Kausar, SSCBS, University of Delhi
Dr. Abhishek Tandon, SSCBS, University of Delhi

Organizing Team

Dr. Gurjeet Kaur, SSCBS, University of Delhi
Dr. Abhimanyu Verma, SSCBS, University of Delhi
Dr. Onkar Singh, SSCBS, University of Delhi
Dr. Kalpna Sagar, KIET Group of Institutes, Ghaziabad
Dr. Purnima Lala Mehta, Assistant Professor, IILM
Dr. Suresh Chavhan, Vellore Institute of Technology, Vellore, India
Dr. Mona Verma, SSCBS, University of Delhi

Preface

We hereby are delighted to announce that Shaheed Sukhdev College of Business Studies, New Delhi, in association with National Institute of Technology Patna, University of Valladolid, Spain, and Korea Institute of Digital Convergence, South Korea, has hosted the eagerly awaited and much coveted International Conference on Innovative Computing and Communication (ICICC-2022) in hybrid mode. The fifth version of the conference was able to attract a diverse range of engineering practitioners, academicians, scholars, and industry delegates, with the reception of abstracts including more than 3600 authors from different parts of the world. The committee of professionals dedicated toward the conference is striving to achieve a high-quality technical program with tracks on innovative computing, innovative communication network and security, and Internet of Things. All the tracks chosen in the conference are interrelated and are very famous among present-day research community. Therefore, a lot of research is happening in the above-mentioned tracks and their related sub-areas. As the name of the conference starts with the word 'innovation,' it has targeted out-of-the-box ideas, methodologies, applications, expositions, surveys, and presentations helping to upgrade the current status of research. More than 850 full-length papers have been received, among which the contributions are focused on theoretical, computer simulation-based research, and laboratory-scale experiments. Among these manuscripts, 200 papers have been included in Springer proceedings after a thorough two-stage review and editing process. All the manuscripts submitted to ICICC-2022 were peer-reviewed by at least two independent reviewers, who were provided with a detailed review proforma. The comments from the reviewers were communicated to the authors, who incorporated the suggestions in their revised manuscripts. The recommendations from two reviewers were taken into consideration while selecting a manuscript for inclusion in the proceedings. The exhaustiveness of the review process is evident, given the large number of articles received addressing a wide range of research areas. The stringent review process ensured that each published manuscript met the rigorous academic and scientific standards. It is an exalting experience to finally see these elite contributions materialize into three book volumes as ICICC-2022 proceedings by Springer entitled *International Conference on Innovative Computing and Communications*. The articles are organized into three volumes in some broad categories covering subject matters on machine learning, data mining, big data, networks, soft computing, and cloud computing, although given the diverse areas of research reported it might not have been always possible.

ICICC-2022 invited five keynote speakers, who are eminent researchers in the field of computer science and engineering, from different parts of the world. In addition to the plenary sessions on each day of the conference, ten concurrent technical sessions are held every day to assure the oral presentation of around 200 accepted papers. Keynote speakers and session chair(s) for each of the concurrent sessions have been leading researchers from the thematic area of the session. A technical exhibition is held during these 2 days of the conference, which has put on display the latest technologies, expositions, ideas, and presentations. The research part of the conference was organized in a total of 42 special sessions and 1 international workshops. These special sessions and international workshops provided the opportunity for researchers conducting research in specific areas to present their results in a more focused environment.

An international conference of such magnitude and release of ICICC-2022 proceedings by Springer has been the remarkable outcome of the untiring efforts of the entire organizing team. The success of an event undoubtedly involves the painstaking efforts of several contributors at different stages, dictated by their devotion and sincerity. Fortunately, since the beginning of its journey, ICICC-2022 has received support and contributions from every corner. We thank them all who have wished the best for ICICC-2022 and contributed by any means toward its success. The edited proceedings volumes by Springer would not have been possible without the perseverance of all the steering, advisory, and technical program committee members.

All the contributing authors owe thanks from the organizers of ICICC-2022 for their interest and exceptional articles. We would also like to thank the authors of the papers for adhering to the time schedule and for incorporating the review comments. We wish to extend our heartfelt acknowledgment to the authors, peer reviewers, committee members, and production staff whose diligent work put shape to ICICC-2022 proceedings. We especially want to thank our dedicated team of peer reviewers who volunteered for the arduous and tedious step of quality checking and critique on the submitted manuscripts. We wish to thank our faculty colleagues Mr. Moolchand Sharma for extending their enormous assistance during the conference. The time spent by them and the midnight oil burnt are greatly appreciated, for which we will ever remain indebted. The management, faculties, administrative, and support staff of the college have always been extending their services whenever needed, for which we remain thankful to them.

Lastly, we would like to thank Springer for accepting our proposal for publishing ICICC-2022 conference proceedings. Help received from Mr. Aninda Bose, the acquisition senior editor, in the process has been very useful.

Delhi, India

Ashish Khanna
Deepak Gupta
Organizers, ICICC-2022

Contents

Editors and Contributors

About the Editors

Dr. Deepak Gupta is an eminent academician and plays versatile roles and responsibilities juggling between lectures, research, publications, consultancy, community service, Ph.D. and post-doctorate supervision, etc. With 13 years of rich expertise in teaching and two years in industry, he focuses on rational and practical learning. He has contributed massive literature in the fields of human–computer interaction, intelligent data analysis, nature-inspired computing, machine learning and soft computing. He has served as Editor-in-Chief, Guest Editor, Associate Editor in SCI and various other reputed journals. He has completed his Post-Doc from Inatel, Brazil, and Ph.D. from Dr. A. P. J. Abdul Kalam Technical University. He has authored/edited 46 books with national/international level publisher (Elsevier, Springer, Wiley, Katson). He has published 180 scientific research publications in reputed international journals and conferences including many SCI Indexed Journals of IEEE, Elsevier, Springer, Wiley and many more. He is the convener and organizer of 'ICICC' springer conference series.

Dr. Ashish Khanna has 17 years of expertise in Teaching, Entrepreneurship and Research and Development. He received his Ph.D. degree from National Institute of Technology, Kurukshetra. He has completed his M.Tech. and B.Tech. GGSIPU, Delhi. He has completed his postdoc from Internet of Things Lab at Inatel, Brazil and University of Valladolid, Spain. He has published around 75 SCI indexed papers in IEEE Transaction, Springer, Elsevier, Wiley and many more reputed Journals. He has around 150 research articles in top SCI/ Scopus journals, conferences and book chapters. He is co-author of around 35 edited and text books. His research interest includes MANET, FANET, IoT, machine learning and many more. He has served the research field as Keynote Speaker/Faculty Resource Person/Session Chair/Reviewer/TPC member/Post-doctorate Supervision. He is Convener and Organizer of 'ICICC' conference series.

Dr. Siddhartha Bhattacharyya (M'10–SM'13) received the bachelor's degree in physics, and the bachelor's and master's degrees in optics and optoelectronics from the University of Calcutta, Kolkata, India, in 1995, 1998 and 2000, respectively, and the Ph.D. degree in computer science and engineering from Jadavpur University, Kolkata, in 2008. He is currently serving as Principal of Rajnagar Mahavidyalaya, Birbhum, India. Prior to this, he was Professor with the Department of Computer Science and Engineering, Christ University, Bengaluru, India. He also served as Principal of the RCC Institute of Information Technology, Kolkata. He served as Senior Research Scientist with the Faculty of Electrical Engineering and Computer Science, VSB Technical University of Ostrava, Ostrava, Czech Republic. He is a full foreign member of the Russian Academy of Natural Sciences. He has co-authored six books, co-edited 76 books and has authored or co-authored more than 300 research publications in international journals and conference proceedings. He holds four patents. His research interests include soft computing, pattern recognition, multimedia data processing, hybrid intelligence, social networks and quantum computing.

Dr. Aboul Ella Hassanien is Founder and Head of the Egyptian Scientific Research Group (SRGE). Hassanien has more than 1000 scientific research papers published in prestigious international journals and over 50 books covering such diverse topics as data mining, medical images, intelligent systems, social networks and smart environment. Prof. Hassanien won several awards including the Best Researcher of the Youth Award of Astronomy and Geophysics of the National Research Institute, Academy of Scientific Research (Egypt, 1990). He was also granted a scientific excellence award in humanities from the University of Kuwait for the 2004 Award and received the superiority of scientific—University Award (Cairo University, 2013). Also, he honored in Egypt as the best researcher in Cairo University in 2013. He was also received the Islamic Educational, Scientific and Cultural Organization (ISESCO) prize on Technology (2014) and received the state award for excellence in engineering sciences 2015.

Dr. Sameer Anand is currently working as Assistant Professor in the Department of Computer science at Shaheed Sukhdev College of Business Studies, University of Delhi, Delhi. He has received his M.Sc., M.Phil., and Ph.D. (Software Reliability) from Department of Operational Research, University of Delhi. He is a recipient of 'Best Teacher Award' (2012) instituted by Directorate of Higher Education, Government of NCT, Delhi. The research interest of Dr. Anand includes operational research, software reliability and machine learning. He has completed an Innovation project from the University of Delhi. He has worked in different capacities in International Conferences. Dr. Anand has published several papers in the reputed journals like *IEEE Transactions on Reliability*, *International Journal of Production Research* (Taylor & Francis), *International Journal of Performability Engineering*, etc. He is a member of Society for Reliability Engineering, Quality and Operations Management. Dr. Sameer Anand has more than 16 years of teaching experience.

Dr. Ajay Jaiswal is currently serving as Assistant Professor in the Department of Computer Science of Shaheed Sukhdev College of Business Studies, University of Delhi, Delhi. He is a co-editor of two books/Journals and co-author of dozens of research publications in International Journals and conference proceedings. His research interest includes pattern recognition, image processing, and machine learning. He has completed an interdisciplinary project titled 'Financial Inclusion-Issues and Challenges: An Empirical Study' as Co-PI. This project was awarded by the University of Delhi. He obtained his masters from the University of Roorkee (now IIT Roorkee) and Ph.D. from Jawaharlal Nehru University, Delhi. He is a recipient of the best teacher award from the Government of NCT of Delhi. He has more than nineteen years of teaching experience.

Contributors

Abirami K. Department of Computer Science, Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, India

Aggarwa Ritu MMICT & BM, MMDU, Mullana, Ambala, Haryana, India

Ahmad Tanvir CSE, Jamia Millia Islamia University, Delhi, India

Akash Guna R. T. Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

Al Ahdal Ahmed Lovely Professional University, Jalandhar, India

Almeer Shaima Shawqi Space Data Analysis Laboratory, National Space Science Agency, Al Hidd, Kingdom of Bahrain

Anbazhagan E. Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Chennai, India

Arora Anuja Department of Computer Science and Engineering and Information Technology, Jaypee Institute of Information Technology, Noida, India

Arora Nitika MMICT & BM, MMDU, Mullana, Ambala, Haryana, India

Ashiq Mahmood Md. Institute of Information and Communication Technology, Khulna University of Engineering and Technology, Khulna, Bangladesh

Ashiqul Amin Md. Northern Arizona University, Flagstaff, AZ, USA

Bagga Manju MMICT & BM, MMDU, Mullana, Ambala, Haryana, India

Balasubramanian Abirami Sri S. Ramasamy Naidu Memorial College, Sattur, India

Bansal Anam GZSCCET, MRSPTU, Bathinda, India

Batra Isha Lovely Professional University, Phagwara, Punjab, 144001 India

Batra Neera Computer Science & Engineering Department, MMEC, Maharishi Markandeshwar (Deemed to Be University), Mullana, Ambala, India

Beena B. M. Department of Computer Science and Engineering, Amrita School of Engineering, Bengaluru, India

Bharath Anuj P. Department of Computer Science and Engineering, Amrita School of Engineering, Coimbatore, Coimbatore, India

Bhardwaj Akashdeep University of Petroleum and Energy Studies, Dehradun, India

Bhaskar Anirudh Centre for Computational Engineering and Networking, Amrita School of Engineering Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, India

Bhatia Manjot K. Jagan Institute of Management Studies, Rohini, Delhi, India

Bhattacharya Supratim Department of Computer Science and Engineering, University of Engineering and Management, Kolkata, India

Bhattacharya Tanmay Department of Information Technology, Techno India, Saltlake, Kolkata, India

Boudlal Hicham Faculty of Sciences, Mohammed First University, Oujda, Morocco

Chakraborty Pinaki Department of Computer Science and Engineering, Netaji Subhas University of Technology, New Delhi, India

Chawla Meenu Maulana Azad National Institute of Technology, Bhopal, India

Chawla Priyanka Lovely Professional University, Jalandhar, India

Chereddy Spandana Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Chennai, India

Chetry Arjun Assam Don Bosco University, Guwahati, India

Chhabra Kritika Ryerson University, Toronto, ON, Canada

Choudhary Anshika Department of Computer Science and Engineering and Information Technology, Jaypee Institute of Information Technology, Noida, India

Choudhury Saumadip Dey Department of Computer Science and Engineering, University of Engineering and Management, Kolkata, India

Chourey Nidhi M. Department of Computer Science and Engineering, Oriental University, Indore, India

Chug Anuradha University School of Information, Communication, and Technology, Guru Gobind Singh Indraprastha University, New Delhi, India

D'Silva Jovi Assam Don Bosco University, Guwahati, Assam, India

Deepthi A. Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, Kerala, India

Dey Apurba Department of Computer Science and Engineering, University of Engineering and Management, Kolkata, India

Dhavale Sunita V. Department of CSE, Defence Institute of Advanced Technology (DIAT), Pune, India

Divyanth C. Department of Computer Science and Engineering, Amrita School of Engineering, Coimbatore, Coimbatore, India

Dornadula Girirajasekhar Department of Pharmacy Practice, Annamacharya College of Pharmacy, Rajampeta, India

Fayez Dipty Fatama North Western University, Khulna, Bangladesh

Garg Naresh Kumar GZSCCET, MRSPTU, Bathinda, India

Gaur Rajkumar ITCA, MMMUT, Gorakhpur, Uttar Pradesh, India

Ghosh Sushmit Department of Computer Science and Engineering, University of Engineering and Management, Kolkata, India

GopaKumar G. Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, Kerala, India

Gourav Kumar University Institute of Computing, Chandigarh University, Chandigarh, India

Goyal Sonali Computer Science & Engineering Department, MMEC, Maharishi Markandeshwar (Deemed to Be University), Mullana, Ambala, India

Gunturu Lakshmi Narasimha Department of Pharmacy Practice, Annamacharya College of Pharmacy, Rajampeta, India

Gupta Dev Delhi Technological University, New Delhi, India

Gupta Harshit Department of Computer Science and Engineering, Netaji Subhas University of Technology, New Delhi, India

Gupta Mayank Maulana Azad National Institute of Technology, Bhopal, India

Harshith N. B. S. P. S. Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, Kerala, India

Hussein Sadoon Department of Physic, College of Science, University of Mosul, Mosul, Iraq

Ippatapu Venkata Srisurya Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Chennai, India

Jadhav Vaibhav Maulana Azad National Institute of Technology, Bhopal, India

Jain Sweta Maulana Azad National Institute of Technology, Bhopal, India

Jamal Mohamed Fuad Amin Mohamed Department of ICT, EDICT, Bahrain Polytechnic, Isa Town, Kingdom of Bahrain

Jeet Singh Amar Department of Computer Science, Himachal Pradesh University, Shimla, India

John Saju P. Department of Computer Science and Engineering, Jyothi Engineering College, Cheruthuruthy, Thrissur, Kerala, India

Joyti Roy Tamal Institute of Information and Communication Technology, Khulna University of Engineering and Technology, Khulna, Bangladesh

Kandelkar Apurva Lovely Professional University, Phagwara, Punjab, India

Karnani Suruchi Amity University, Gwalior, Madhya Pradesh, India

Kaur Amanpreet Chitkara University Institute of Engineering & Technology, Chitkara University, Rajpura, Punjab, India

Kaur Bavneet Sri Guru Granth Sahib World University, Fatehgarh Sahib, India

Kaur Gaganpreet Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab, India;
Thapar Institute of Engineering and Technology, Patiala, India

Kaushik Sunil University of Petroleum and Energy Studies, Dehradun, India

Khajuria Rakshit Lovely Professional University, Phagwara, Punjab, India

Khanum Saba Assistant Professor, MSIT, New Delhi, India

Kishk Ahmed F. EED, Alexandria Higher Institute of Engineering and Technology (AIET), Alexandria, Egypt

Komalavalli C. Jagan Institute of Management Studies, Rohini, Delhi, India

Krishnasamy Krishnaveni Sri S. Ramasamy Naidu Memorial College, Sattur, India

Kumar Aditya Department of Computer Science and Engineering, Netaji Subhas University of Technology, New Delhi, India

Kumar Mukesh Department of Computer Science, Himachal Pradesh University, Shimla, India

Kumar Goyal Raman Thapar Institute of Engineering and Technology, Patiala, India

Lalithamani N. Department of Computer Science and Engineering, Amrita School of Computing, Amrita Vishwa Vidyapeetham, Coimbatore, India

Laroiya Chetna Jagan Institute of Management Studies, Rohini, Delhi, India

Lavanya B. Department of Computer Science, University of Madras, Chennai, India

Lokesh Sai Phani Babu S. Department of Computer Science and Engineering, Amrita School of Engineering, Coimbatore, Coimbatore, India

Madan Suman Jagan Institute of Management Studies, Rohini, Delhi, India

Mahalakshmi R. Department of Computer Science and Engineering, Amrita School of Computing, Amrita Vishwa Vidyapeetham, Coimbatore, India

Majumdar Stuti Department of Computer Science and Engineering, Brainware University, Barasat, Kolkata, India

Malhotra Manisha UIC, Chandigarh University, Mohali, India

Malhotra Ruchika Delhi Technological University, New Delhi, India

Malik Arun Lovely Professional University, Phagwara, Punjab, India

Masood Sarfaraz CSE, Jamia Millia Islamia University, Delhi, India

Mehra Monika Department of ECE Engineering, RIMT University, Mandi Gobindgarh, Punjab, India

Mehta Rajesh Thapar Institute of Engineering and Technology, Patiala, India

Mithillesh Kumar P. Department of Computer Science and Engineering, Amrita School of Engineering, Bengaluru, Amrita Vishwa Vidyapeetham, Bengaluru, India

Mitra Shovon Brac University, Dhaka, Bangladesh

Mittal Prabhat Department of Commerce, Satyawati College (Evening), University of Delhi, Delhi, India

Mohanta Aninda Institute of Engineering and Management, Kolkata, India

Mokhtar Mohamed-Amr A. EED, Alexandria Higher Institute of Engineering and Technology (AIET), Alexandria, Egypt

More Chaitali Fr. Agnel College of Arts and Commerce, Pilar, Goa, India

Mukesh K. Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Chennai, India

Mukherjee Rajendrani Department of Computer Science and Engineering, University of Engineering and Management, Kolkata, India

Munesh Patel Shreyans Computer Engineering MPSTME, NMIMS, Mumbai, Maharashtra, India

Mustafa Khurram Research Scholar, JMI, Okhla, India

Naleer H. M. M. Department of Computer Science, Faculty of Applied Sciences, South Eastern University of Sri Lanka, Oluvil, Sri Lanka

Nanade Archana Computer Engineering MPSTME, NMIMS, Mumbai, Maharashtra, India

Nandy Sanjukta Department of Computer Science and Engineering, University of Engineering and Management, Kolkata, India

Nanthini M. Department of Computing Technologies, SRM Institute of Science and Technology, Kattankulathur, Chennai, Tamil Nadu, India

Narain Panday Akash Department of Computer Science and Engineering, University of Engineering and Management, Kolkata, India

Naveen Nimbagal Raghavendra Department of Pharmaceutics, Sri Adichunchanagiri College of Pharmacy, B.G Nagara, Karnataka, India

Niveditha P. S. APJ Abdul Kalam Technological University, Thiruvananthapuram, Kerala, India

Oviya I. R. Department of Sciences, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Chennai, India

Pamayyagari Kalpana Department of Pharmacy Practice, Annamacharya College of Pharmacy, Rajampeta, India

Parveiz Bazila Department of ECE Engineering, RIMT University, Mandi Gobindgarh, Punjab, India

Patel Cdr Devraj Department of CSE, Defence Institute of Advanced Technology (DIAT), Pune, India

Phonsa Gurbakash Mewar University, Gangarar, India

Pradeep Mohan Kumar K. Department of Computing Technologies, SRM Institute of Science and Technology, Kattankulathur, Chennai, Tamil Nadu, India

Prakash Shiva ITCA, MMMUT, Gorakhpur, Uttar Pradesh, India

Prakash Tiwari Sudhanshu Mewar University, Gangarar, Rajasthan, India

Prashant Shah Dhruv Computer Engineering MPSTME, NMIMS, Mumbai, Maharashtra, India

Prashanth V. R. Centre for Computational Engineering and Networking, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, Tamil Nadu, India

Pulari Sini Raj Department of ICT, EDICT, Bahrain Polytechnic, Isa Town, Kingdom of Bahrain

Radhika G. Department of Computer Science, Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, India

Raghava Reddy Ch. Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, Kerala, India

Rahul K. Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

Raj Anu ITCA, MMMUT, Gorakhpur, Uttar Pradesh, India

Rajiv Kumar Bhagat Shubh Computer Engineering MPSTME, NMIMS, Mumbai, Maharashtra, India

Rajwal Swati Division of Computer Engineering, Netaji Subhas University of Technology, New Delhi, India

Ram Kodumuri Bhargav BVRIT HYDERABAD College of Engineering for Women, Hyderabad, Hyderabad, Telangana, India

Rama Sudha K. Department of Electrical Engineering, Andhra University College of Engineering(A), Visakhapatnam, Andhra Pradesh, India

Ramakrishnan Rohith Centre for Computational Engineering and Networking, Amrita School of Engineering Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, India

Rani Samta CSE, Jamia Millia Islamia University, Delhi, India

Rathee Sonia Maharaja Surajmal Institute of Technology, New Delhi, India

Ravi Shanker Reddy T. Department of Computer Science and Engineering, Amrita School of Engineering, Bengaluru, India

Revathi M. Department of Electrical Engineering, Andhra University College of Engineering(A), Visakhapatnam, Andhra Pradesh, India

Roy Diti Institute of Information and Communication Technology, Khulna University of Engineering and Technology, Khulna, Bangladesh

Saanidhi Delhi Technological University, New Delhi, India

Sabitha Banu A. Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, Tamil Nadu, India

Sachin Kumar S. Centre for Computational Engineering and Networking, Amrita School of Engineering Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, Tamil Nadu, India

Saha Suprativ Department of Computer Science and Engineering, Brainware University, Barasat, Kolkata, India

Sahu Priyanka University School of Information, Communication, and Technology, Guru Gobind Singh Indraprastha University, New Delhi, India

Sami Ahmed Department of Computer Science, College of Computer Science and Mathematics, University of Mosul, Mosul, Iraq

Saritha S. K. Maulana Azad National Institute of Technology, Bhopal, India

Serrhini Mohammed Faculty of Sciences, Mohammed First University, Oujda, Morocco

Shakya Harish Kumar Department of Computer Science and Engineering, Amity University, Gwalior, Madhya Pradesh, India;
Thapar Institute of Engineering and Technology, Patiala, Punjab, India

Shalu Maharaja Surajmal Institute of Technology, New Delhi, India

Shanmuga Sundari M. BVRIT HYDERABAD College of Engineering for Women, Hyderabad, Hyderabad, Telangana, India

Sharma Anuj Lovely Professional University, Phagwara, Punjab, India

Sharma Arvind K. IIT Mandi iHUB and HCI Foundation, Mandi, Himachal Pradesh, India;
Department of Computer Science & Engineering, University School of Engineering & Technology, Rayat Bahra University, Mohali, Punjab, India

Sharma Ashok University of Jammu, Jammu, India

Sharma Kapil Maulana Azad National Institute of Technology, Bhopal, India

Sharma Shabnam CMR University, Bengaluru, Karnataka, India

Sharma Uzzal Department of Computer Applications, School of Technology, Assam Don Bosco University, Guwahati, Assam, India

Shibly F. H. A. Assam Don Bosco University, South Eastern University of Sri Lanka, Oluvil, Sri Lanka

Sikarwar Ranjana Department of Computer Science and Engineering, Amity University, Gwalior, Madhya Pradesh, India;
Thapar Institute of Engineering and Technology, Patiala, Punjab, India

Sikha O. K. Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

Simpson Serin V. Department of Computer Science and Engineering, SCMS School of Engineering and Technology, Ernakulam, Kerala, India

Sindhuja D. Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, Kerala, India

Singh Amit Prakash University School of Information, Communication, and Technology, Guru Gobind Singh Indraprastha University, New Delhi, India

Singh Dinesh Division of Plant Pathology, Indian Agricultural Research Institute, New Delhi, India

Singh Gurpreet CSE, Punjab Institute of Technology, Rajpura (MRSPTU, Bathinda), India

Singh Parveen Cluster University, Jammu, India

Singh Ravinder Pal Department of ECE Engineering, RIMT University, Mandi Gobindgarh, Punjab, India

Singh Shashank Sheshar Department of Computer Science and Engineering, Amity University, Gwalior, Madhya Pradesh, India;
Thapar Institute of Engineering and Technology, Patiala, Punjab, India

Sivasubramanian Arrun Centre for Computational Engineering and Networking, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, Tamil Nadu, India

Soman K. P. Centre for Computational Engineering and Networking, Amrita School of Engineering Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, Tamil Nadu, India

Soni Rashmi Department of Computer Science and Engineering, Oriental University, Indore, India

Sri Ranganath U. Department of Computer Science and Engineering, Amrita School of Engineering, Coimbatore, Coimbatore, India

Srivastva Jyoti ITCA, MMMUT, Gorakhpur, Uttar Pradesh, India

Sudha Rani M. BVRIT HYDERABAD College of Engineering for Women, Hyderabad, Hyderabad, Telangana, India

Supriya M. Department of Computer Science and Engineering, Amrita School of Engineering, Bengaluru, Amrita Vishwa Vidyapeetham, Bengaluru, India

Tahiri Ahmed Faculty of Sciences, Mohammed First University, Oujda, Morocco

Tayel Mazhar B. EED, Faculty of Engineering, Alexandria University, Alexandria, Egypt

Thasin Abida Department of Physic, College of Science, University of Mosul, Mosul, Iraq

Tiwari Namita Maulana Azad National Institute of Technology, Bhopal, India

Tuli Krishan UIC, Chandigarh University, Mohali, India

Vadakedath Anirudh Centre for Computational Engineering and Networking, Amrita School of Engineering Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, India

Vageeswari U. Department of Computer Science, University of Madras, Chennai, India

Vinay Tailor Jainam Computer Engineering MPSTME, NMIMS, Mumbai, Maharashtra, India

Wadhawan Savita MMICTBM (MCA), MMDU, Ambala, Haryana, India;
Department of Computer Science and Engineering, Punjabi University, Patiala, Punjab, India

Yadav Amita Maharaja Surajmal Institute of Technology, New Delhi, India

Yadav Savita Department of Computer Science and Engineering, Netaji Subhas University of Technology, New Delhi, India

Zafar Sherin Jamia Hamdard, New Delhi, India

Constructing Interval Type-2 Fuzzy Systems (IT2FS) with Memetic Algorithm: Elucidating Performance with Noisy Data

Savita Wadhawan and Arvind K. Sharma

Abstract Fuzzy modeling is a challenging task and becomes more complex when designing T2FS, which requires identification of more parameters as compared to T1FS. The problem of fuzzy modeling can be expressed as a high-dimensional search and optimization process, and EAs have the ability to search for optimal solutions in high-dimensional search space, so researchers used various EAs for fuzzy modeling. GAs are widely used for finding solutions in large search spaces, and MAs have characteristics of both global and local optimizations. This paper describes how to use MAs and GAs to identify IT2FS, including how to build MFs for both input and output, as well as how to generate a rule base from a data collection. The efficiency of T1FS and IT2FS for noisy data is also compared with GAs and MAs in the paper. For comparison, we consider four different problems: a rapid Ni–Cd battery charger, data from Box and Jenkins's gas furnace, and the iris and wine classification datasets. In the presence of noise, the results imply that IT2FS is more efficient than T1FS, and MAs are more efficient than GAs.

Keywords Fuzzy modeling · T1FS · IT2FS · Memetic algorithm (MAs) · Genetic algorithm (GAs)

S. Wadhawan (✉)
MMICTBM (MCA), MMDU, Mullana, Ambala, Haryana, India
e-mail: savitawadhawan@mmumullana.org

Department of Computer Science and Engineering, Punjabi University, Patiala, Punjab, India

A. K. Sharma
IIT Mandi iHUB and HCI Foundation, Mandi, Himachal Pradesh, India
e-mail: arvind@ihubiitmandi.in

Department of Computer Science & Engineering, University School of Engineering & Technology, Rayat Bahra University, Mohali, Punjab, India

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_1

1 Introduction

Fuzzy logic-based systems have their broad applications in the field of intelligent control, signal processing, approximate reasoning and prediction, etc. Fuzzy systems are classified as type-1 fuzzy logic-based systems (T1FS) and type-2 fuzzy logic-based systems (T2FS). Fuzzy logic-based systems are proved to be the best in dealing with uncertainty, but the uncertainty in T1FS is lost during inference process because of the use of type-1 fuzzy sets as membership functions with assigned degree within an interval [0,1] [24]. On the contrary, T2FS deals by using type-2 fuzzy sets in which associated membership degree is expressed as uncertain type-1 fuzzy sets [2–35]. Based upon simplicity and efficiency, the most extensively used version of T2FS is interval type-2 fuzzy logic-based systems (IT2FS) used in various applications like decision making [6–46], time-series forecasting [10–28], clustering [13–50], control [17–30], wireless communication [20–41], health-care/smart care devices [23–15], robotics [2, 19], etc.

Multiple factors are responsible in designing a fuzzy logic-based system, and it includes choice of membership functions, rule base generation [45], inference mechanism consisting (composition, selection, and aggregation operators), and defuzzification strategy. Knowledge-driven and data-driven approaches are used in designing a fuzzy system. In knowledge-driven strategy, factors are completely decided by domain expert, which may cause difficulty either due to incomplete knowledge or because of large problem space. Without any prior knowledge, numeric information obtained from available input–output data is used in designing fuzzy systems in data-driven strategy [37]. The design and identification of fuzzy logic base systems can be described as an optimization and search problem in a high-dimensional space, with each point reflecting the behavior of the system. System performance with taken objective/fitness functions forms hypersurface, and optimal fuzzy logic-based system designing is equivalent as finding optimal locations on this hypersurface [23]. Evolutionary algorithms are best to use while searching in hypersurface and also used to learn the parameters of fuzzy logic-based systems. Study reveals successful applications of evolutionary algorithms in type-2 fuzzy modeling with particle swarm optimization(PSO) [31–8], ant colony optimization(ACO) [35–51], genetic algorithms(GAs) [6, 40–42], etc. Hybridized methods are also proposed by researchers for the evolution of type-2 fuzzy systems [43–48]. Among different evolutionary algorithms, MAs have the characteristics of both global and local search. Global search is performed by traditional evolutionary algorithm, and a local search is used to enhance the individuals of population. This hybridization combines the exploration capabilities of evolutionary algorithm, and the exploitation capability of local search, hence, performs better. To the best in our knowledge, a few researchers work for type-2 fuzzy modeling with memetic algorithms (MAs) [3].

Contribution of the Paper:

This paper introduced an effective methodology to generate interval type-2 fuzzy system (IT2FS) from data using EAs. GAs are best to find optimal solution in large search space, and MAs are advantageous due to hybridization of local search and global search; this paper compares the efficiency of these two for modeling of T1FS and T2FS. Because type-2 fuzzy systems are known for their ability to deal with uncertainty and noise, noise was added into datasets, and the performance of type-1 fuzzy-based systems (T1FS) and interval type-2 fuzzy-based systems (IT2FS) with this noisy data was presented.

Further, Sect. 2 of the paper presents brief introduction about type-2 fuzzy logic systems followed by introduction of GAs and MAs in Sect. 3. Section 4 describes the MAs-based mathematical modeling of type-2 fuzzy-based system which is followed by the experimental study with four benchmark datasets in Sect. 5, and concluding remarks based on present study are discussed in Sect. 6

Motivation for Work

Literature study reveals that different evolutionary algorithms are used by different researchers for fuzzy modeling like PSO [31–32], GAs [6, 40–42], BB-BC [36–49], and ACO [18, 51]. To the best in our knowledge, MAs were used for learning the parameters of interval type-2 fuzzy rule-based system by Acampora et al. [3]. GAs are best known in EAs to find optimal solution and the reported advantages of MAs, which are due to hybridization of local search into evolutionary algorithms, and the capability of type-2 fuzzy-based systems to handle uncertainty and noise in data makes realization for this study. Keeping this in view, a methodology is introduced for interval type-2 fuzzy logic-based systems (IT2FS) with MAs and presented experimental study after introducing noise in the data.

2 Introduction to Type-2 Fuzzy Logic System

Type-2 fuzzy logic systems are capable of handling uncertainties. The architecture of type-2 fuzzy logic system is presented in Fig. 1 consisting of various modules, viz.

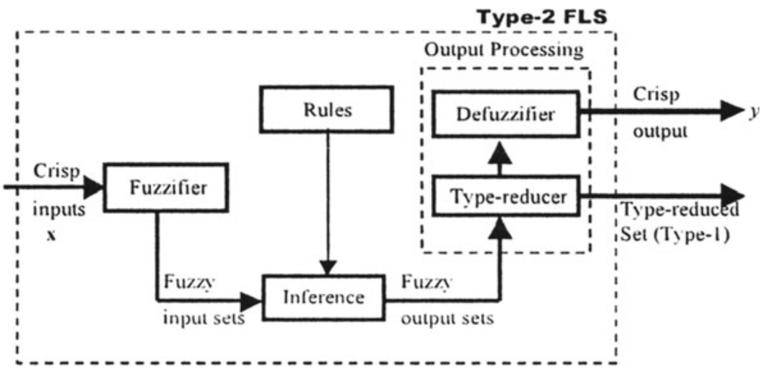


Fig. 1 Type-2 fuzzy systems

fuzzifier, inference, and output processing box. Fuzzifier converts the crisp inputs into fuzzy inputs. Inference mechanism derives the fuzzy output from fuzzy inputs by applying a fuzzy reasoning process. The output processing mechanism is the combination of type reducer and defuzzifier. Type reducer converts the type-2 fuzzy output into type-1 fuzzy output. A number of algorithms are presented in the literature for type reduction. Here, we use Karnik–Mendal, i.e., KM algorithm is used for type reduction. Type-1 fuzzy output is then translated into a crisp output by defuzzifier.

3 Memetic Algorithms

MAs are categorized as stochastic global search algorithms, allowing individuals to gain some experience through local search before going in the evolutionary process [36]. MAs are similar to evolutionary algorithms, with the exception that each member of the population is subjected to a local search to enhance it, resulting in a population of local optimum solutions. The key feature of MAs is to combine the exploration capabilities of evolutionary algorithm, and the exploitation capability of local search makes it better. But a number of issues are addressed while designing MAs like (i) frequency of local search means how frequently the local search be applied? (ii) Which individual from population, improved by local search? (iii) At which time, the local search be applied? (iv) What would be the computational efforts allowed to each local search? [25]. To address all of these concerns in our fuzzy system design, we include two parameters: local search probability and interval.

4 Mathematical Modeling

The fuzzy model identification problem for type-2 fuzzy models can be described as a search and optimization problem in a high-dimensional space, where each point represents the system behavior by providing information about fuzzy structure such as membership functions, rule base, and so on. Because these are extremely complicated issues, evolutionary algorithms are capable of finding the best or near-best solution in a given complex search space. Because of their parallel nature, these algorithms can find global optimum solutions without knowing anything about search space characteristics. The following principles apply to fuzzy model identification for type-2 fuzzy systems utilizing evolutionary algorithms:

I. Representation of complete type-2 fuzzy systems in one solution, i.e., chromosome which requires encoding mechanism
II. Defining an objective function to evaluate the performance of type-2 fuzzy chromosomal models. The performance of the chromosome is evaluated using the mean square error (MSE) given in Eq. 1.

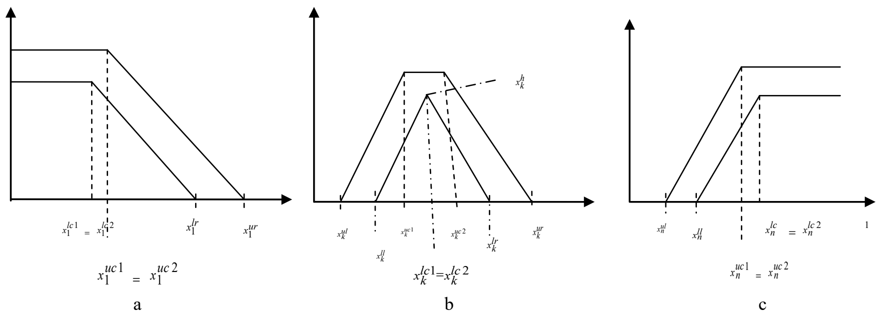


Fig. 2 **a** Zed type MF, **b** triangular type MF, and **c** sigma type MF

$$\text{MSE} = \frac{1}{N}\sum_{k=1}^{N}\left[z(k) - z(k)'\right]^2 \tag{1}$$

where $z(k)$ and $z(k)$' represent the desired output and actual output of the fuzzy system. Lower the value of the MSE, better will be the system. N represents number of data points taken for system validation.

Encoding Mechanism

Multi-input single-output (MISO) fuzzy model is considered for the modeling of type-2 fuzzy system with some constraints like (i) the model should work with fixed number of triangular type membership functions for both input and output variables considering zed and sigma type for first and last membership functions. (ii) Some defined constraints should be ensured for the overlapping between adjacent membership functions. (iii) The system should work with complete rule base.

Encoding Method for Membership Functions

Consider the membership functions zed, sigma, and triangular, as given in Fig. 2. Because zed and sigma type membership functions are always be first and last, MF, therefore, is represented by x_1 and x_n. All the intermediate triangular type membership functions are represented by x_k. Considering the example of triangular type membership function, $x_k^{ul}, x_k^{uc1}, x_k^{uc2}\, x_k^{ur}$ represents left anchor, cortex points, and right anchor of upper membership function and $x_k^{ll}, x_k^{lc1}, x_k^{lc2}\, x_k^{lr}$ represents left anchor, cortex points, and right anchor of lower membership function. x_k^h gives height of lower membership function.

Thus, parameters for one membership function (MF) are $x_k^{ul}, x_k^{ll}, x_k^{uc1}, x_k^{lc1}, x_k^{lc2}, x_k^{uc2}, x_k^{lr}, x_k^{ur}$ (eigth parameters) and x_k^h(height of LMF).

Constraints on parameters for one membership function (MF) are as follows:

$$x_k^{ul} < x_k^{ll} < x_k^{uc1} < x_k^{lc1} = x_k^{lc2} < x_k^{uc2} < x_k^{lr} < x_k^{ur} \text{ and } x_k^h$$

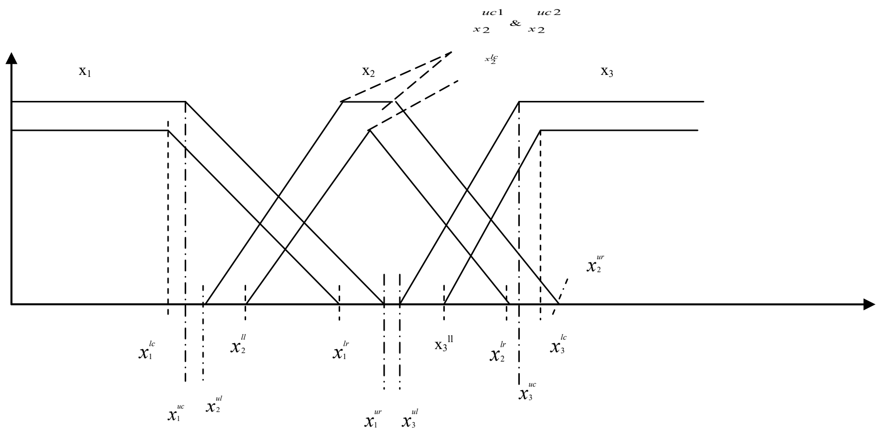


Fig. 3 Representation of overlapping between three membership functions of a variable with constraints

Because $x_k^{lc1} = x_k^{lc2}$, therefore a number of parameters are 7 and another one is height of LMF, i.e., x_k^h.

Consider a variable with three fuzzy sets as shown in Fig. 3 to impose the additional constraints that ensure the overlapping between adjacent membership functions.

For input:$x_{\min} = x_1^{ul} = x_1^{ll}$ and $x_1^{uc1} = x_1^{uc2}$ and $x_1^{lc1} = x_1^{lc2}$

$$\begin{array}{ll} \text{For input} & x_{\min} = x_1^{ul} = x_1^{ll} \text{ and } x_1^{uc1} = x_1^{uc2} \text{ and } x_1^{lc1} = x_1^{lc2} \\ & x_{\max} = x_n^{ul} = x_n^{ll} \text{ and } x_n^{uc1} = x_n^{uc2} \text{ and } x_n^{lc1} = x_n^{lc2} \\ \text{For output} & y_{\min} = y_1^{ul} = y_1^{ll} \text{ and } y_1^{uc1} = y_1^{uc2} \text{ and } y_1^{lc1} = y_1^{lc2} \\ & y_{\max} = y_n^{ul} = y_n^{ll} \text{ and } y_n^{uc1} = y_n^{uc2} \text{ and } y_n^{lc1} = y_n^{lc2} \end{array}$$

where $x_{\min}, x_{\max}$ are the minimum and maximum value of the input variable and $y_{\min}$, $y_{\max}$ are the minimum and maximum value for the output variable, respectively.

Additional constraints that ensure the overlapping between adjacent membership functions can be represented as below:

$$\begin{aligned} x_{\min} \le x_1^{lc1} = x_1^{lc2} \le x_1^{uc1} = x_1^{uc2} \le x_2^{ul} < x_2^{ll} < x_1^{lr} < x_1^{ur} < x_2^{uc1} < x_2^{lc1} < x_2^{uc2} \\ < x_3^{ll} < x_2^{lr} < x_2^{ur} \le x_3^{uc1} = x_3^{uc2} \le x_3^{lc1} = x_3^{lc2} \le x_{\max} \end{aligned}$$

The aforementioned additional constraint can be applied to any number of membership functions and input/output variables. Let $m_1,\ m_2, \ldots m_n$ be the number of membership functions for input1, input2 ,... and input n. t be the number of membership functions for output variable. Therefore, each input variable is represented as x_{im_k} and is read as kth membership function of ith input variable. Constraints on each input/output variables of a fuzzy model are represented in Fig. 4.

The number of parameter to be optimized in one MF is = 7

The number of membership functions (MFs) for ith input = m_i

Input / Output Variable.	Constraints on each input and output variables.
Input variable 1	$x_{1\min} \le x_{11}^{lc1} = x_{11}^{lc2} \le x_{11}^{uc1} = x_{11}^{uc2} \le x_{12}^{ul} < x_{12}^{ll} < x_{11}^{lr} < x_{11}^{ur} < x_{12}^{uc1} < x_{12}^{lc1} < x_{12}^{uc2} < x_{13}^{ll}$ ……………………… $< x_{1m_1}^{ul} < x_{1m_1}^{ll} < x_{1(m_1-1)}^{lr} < x_{1(m_1-1)}^{ur} < x_{1m_1}^{uc1} = x_{1m_1}^{uc2} \le x_{1m_1}^{lc1} = x_{1m_1}^{lc2} \le x_{1\max}$
Input variable 2	$x_{2\min} \le x_{21}^{lc1} = x_{21}^{lc2} \le x_{21}^{uc1} = x_{21}^{uc2} \le x_{22}^{ul} < x_{22}^{ll} < x_{21}^{lr} < x_{21}^{ur} < x_{22}^{uc1} < x_{22}^{lc1} < x_{22}^{uc2} < x_{23}^{ll}$ ……………………… $< x_{2m_2}^{ul} < x_{2m_2}^{ll} < x_{2(m_2-1)}^{lr} < x_{2(m_2-1)}^{ur} < x_{2m_2}^{uc1} = x_{2m_2}^{uc2} \le x_{2m_2}^{lc1} = x_{2m_2}^{lc2} \le x_{2\max}$

Input variable n	$x_{n\min} \le x_{n1}^{lc1} = x_{n1}^{lc2} \le x_{n1}^{uc1} = x_{n1}^{uc2} \le x_{n2}^{ul} < x_{n2}^{ll} < x_{n1}^{lr} < x_{n1}^{ur} < x_{n2}^{uc1} < x_{n2}^{lc1} < x_{n2}^{uc2} < x_{n3}^{ll}$ ……………………… $< x_{nm_n}^{ul} < x_{nm_n}^{ll} < x_{n(m_n-1)}^{lr} < x_{n(m_n-1)}^{ur} < x_{nm_n}^{uc1} = x_{nm_n}^{uc2} \le x_{nm_n}^{lc1} = x_{nm_n}^{lc2} \le x_{n\max}$
Output variable	$y_{\min} \le y_1^{lc1} = y_1^{lc2} \le y_1^{uc1} = y_1^{uc2} \le y_2^{ul} < y_2^{ll} < y_1^{lr} < y_1^{ur} < y_2^{uc1} < y_2^{lc1} < y_2^{uc2} < y_3^{ll}$ ……………………… $< y_t^{ul} < y_t^{ll} < y_{t-1}^{lr} < y_{t-1}^{ur} < y_t^{uc1} = y_t^{uc2} \le y_t^{lc1} = y_y^{lc2} \le y_{\max}$

Fig 4 Representation of constraints on input/output variables of a fuzzy model

Total numbers of parameters for each input is $7m_i$

For first and last MF, there are only four parameters to be optimized. As a result, the minimum number of parameters required to encode the membership functions for each variable is 7 m_i—(2 * 3) =7 m_i −6.

Therefore, the total number of parameters representing n inputs is

$$\sum_{i=1}^{n} 7\,m_i - 6 \tag{2}$$

The number of required parameters for output is

$$7t - 6 \tag{3}$$

Chromosome size for membership functions (from Eqs (2) and (3))

$$= \left(\sum_{i=1}^{n} 7\,m_i - 6 \right) + (7t - 6) \tag{4}$$

Considering the optimization of height of LMF of each membership function for each input variable, size is different, given in Fig. 5.

Then total size of one chromosome is (when height is to be optimized from Eqs. (4) and (5)):

$$\left(\sum_{i=1}^{n} 7\,m_i - 6 \right) + (7t - 6) + \left(\sum_{i=1}^{n} m_i + t \right) = \left(\sum_{i=1}^{n} 8\,m_i - 6 \right) + (8t - 6) \tag{6}$$

Input variable	Representation (Height)	Total no. of Parameters
Input variable 1	$\overset{h}{x}_{11}, \overset{h}{x}_{12}, \ldots\ldots\ldots\ldots\ldots, \overset{h}{x}_{1m_1}$	$m1$
Input variable 2	$\overset{h}{x}_{21}, \overset{h}{x}_{22}, \ldots\ldots\ldots\ldots\ldots, \overset{h}{x}_{2m_2}$	$m2$

Input variable n	$\overset{h}{x}_{n1}, \overset{h}{x}_{n2}, \ldots\ldots\ldots\ldots\ldots, \overset{h}{x}_{nm_n}$	mn
Output variable	$\overset{h}{y}_1, \overset{h}{y}_2, \ldots\ldots\ldots\ldots\ldots, \overset{h}{y}_t$	t
Total Size		$\sum_{i=1}^{n} mi + t$ (5)

Fig 5 Representing size after considering height

Encoding method for rule base:

We included the full rule base when designing the system, the required size to represent the entire rule base is given as follows:

Chromosome size (for rule base) =

$$\prod_{i=1}^{n} m_i \tag{7}$$

Simply add Eqs. 6 and 7 together to determine the number of parameters needed to encode the system exhibiting chromosomal size.

Chromosome size of fuzzy model $= \left(\sum_{i=1}^{n} 8\, m_i - 6\right) + (8t - 6) + \prod_{i=1}^{n} m_i$

or is given as

$$\left(\sum_{i=1}^{n+1} 8\, m_i - 6\right) + \prod_{i=1}^{n} m_i \tag{8}$$

Equation 8 gives the size of the chromosome, which consists of membership function parameters for each input variable, output variable, and rule base corresponding to the fuzzy model.

Figure 6 shows the framework for type-2 fuzzy model identification using an evolutionary method, whereas Fig. 7 shows the pseudocode.

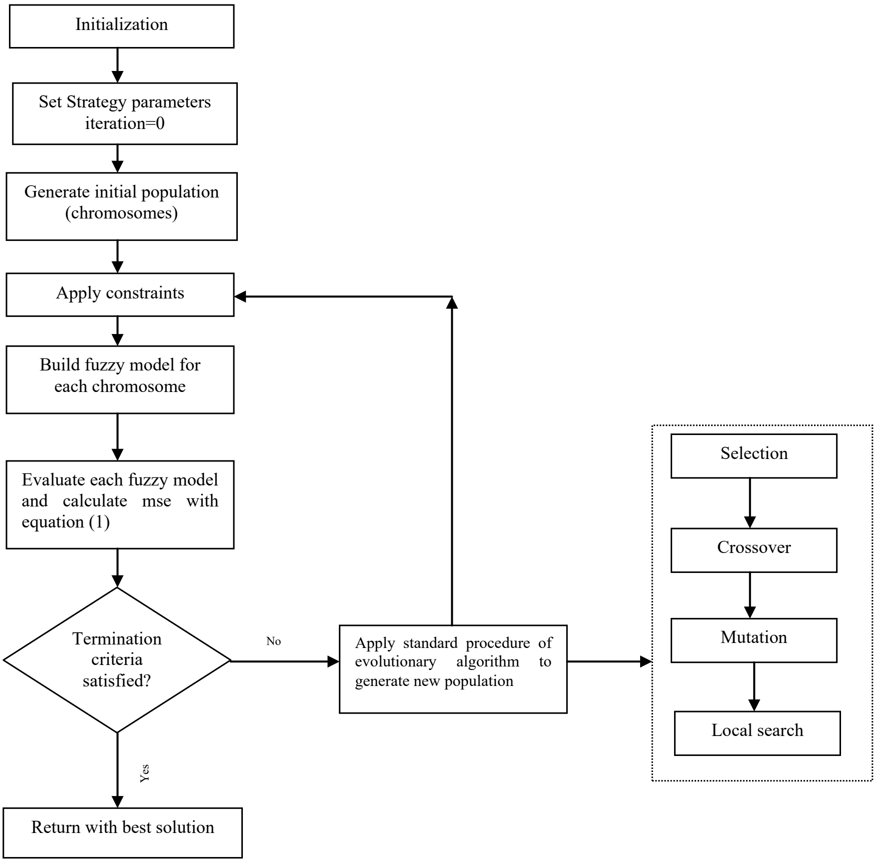


Fig. 6 Framework for fuzzy model identification trough evolutionary algorithm

```
Begin;
        Define the evolutionary algorithm's strategy parameters;
        Iteration = 0;
        Generate initial population of chromosomes;
        While itr ≤ Max_itr
                Constraint chromosomes;
                For each chromosome, create a fuzzy model.
                Evaluate each fuzzy model and compute MSE using equation (1);
                To obtain the new population, follow the conventional evolutionary algorithm approach. ;
                Itr = itr + 1;
        End
End
```

Fig. 7 Pseudocode for fuzzy model identification through evolutionary algorithm

5 Experimental Study

In this section, we demonstrate the fuzzy modeling for type-1 fuzzy systems (T1FS) and interval type-2 fuzzy systems (IT2FS) for various examples. The framework for type-1 fuzzy modeling was taken as proposed by Wadhawan et al [44]. The evolutionary algorithms GAs and MAs coding have been done in MATLAB 2016a and executed on intel(R) CORE(TM) i5 7200U @ 2.70 GHz processor with 4 GB RAM laptop. Table 1 shows various parameters used for the experimental study with GAs and MAs. In memetic algorithms, the local search probability and interval terms are utilized to balance the high computing cost of construction and to optimize approximations at each generation. As type-2 fuzzy logic systems are best in dealing with uncertain and noisy data, the white Gaussian noise was added using awgn() function of MATLAB. For the efficiency comparison of T1FS and IT2FS, the data is divided into training data and testing data. 30% of the complete dataset is taken for training, while rest of the data is used for testing. For performance evaluation, mean square error (MSE) is to be considered. The result of ten trials, every trial consisting average of ten runs, for each dataset has been presented. To evaluate the performance of this approach, four different problems have been considered.

Experiments on First Two Datasets:

The first rapid Ni–Cd battery charger dataset consists of inputs temperature [T], temperature gradient [dT/dt], and output charging rate (Ct). The purpose for developing this charger was the possible quick charge of the batteries without any breakage to these [23]. The universe of discourse for temperature, temperature gradient, and charging current is [0 50], [0 1], and [0 8], respectively. Ni–Cd battery charger data consists of 561 data points available at http://www.research.4t.com. Table 2 shows the results obtained for GAs and MAs for both T1FS and IT2FS.

Second, the gas furnace dataset by Box and Jenkin is a time-series dataset with a single input of gas flow rate u(t) and a single output of CO_2 concentration y(t). More information about the dataset can be found at https://openmv.net/info/gas-furnace. Sugeno and Yasukava [43] assume ten input variables, y(t−1),...,y(t−4),u(t−1),...,u(t−6), as input candidates to influence the output y(t).

Table 1 Strategy parameters for MAs and GAs

Parameter	MAs	GAs
Population size	20	20
Crossover rate	0.8	0.8
Mutation probability	0.09	0.09
Generation gap	0.7	0.7
Local search probability	0.9	–
Interval	10	–
No. of iterations	2000	2000

Table 2 Experimental results of T1FS and IT2FS for Ni–Cd battery charger dataset

Testing performance in terms of MSE for Ni–Cd battery charger dataset. (Each trial consists of average of ten runs)				
	Genetic algorithm		Memetic algorithm	
Trial	T1FS	IT2FS	T1FS	IT2FS
1	0.5964	0.3932	0.5118	0.2896
2	0.4578	0.3245	0.4436	0.2145
3	0.5595	0.3467	0.5126	0.2995
4	0.4569	0.3125	0.4326	0.2698
5	0.4967	0.3296	0.4598	0.2163
6	0.4898	0.3596	0.3987	0.2756
7	0.4796	0.2889	0.4218	0.3012
8	0.4562	0.3261	0.4325	0.2375
9	0.5123	0.3205	0.4187	0.2123
10	0.4473	0.3084	0.4254	0.2489

Although the original dataset contains 296 data pairs, only 290 of them can be used with the above settings.

Results and Discussions

This section presents the results obtained for the abovementioned datasets along with efficiency comparision of T1FS versus IT2FS and GAs versus MAs. Table 2 shows the results of rapid Ni–Cd battery charger dataset, and Table 4 presents the average, best, and worst case performance in terms of MSE for T1FS and IT2FS with both algorithms. The average performance of T1FS and IT2FS as observed with GAs is 0.49525 and 0.331, respectively, whereas with MAs is 0.44575 and 0.25652. The best and worst case performance can also be compared. Table 3 presents results for gas furnace dataset for ten trials, and Table 5 shows their average, best, and worst case analysis. The average case performances for this dataset with GAs for T1FS and IT2FS are 0.48002 and 0.29133. However, with MAs, the performances of T1FS and IT2FS are 0.39525 and 0.22547, respectively. Analysis of Tables 4 and 5 shows better performance of IT2FS as compared to T1FS with both the algorithms. The performance comparison of two algorithms shows better results with MAs than GAs. For the statistical validation of the results, Paired sample t-test was conducted over MAs and GAs as well as over T1FS and IT2FS at the significant level of 5%. The paired sample t-test is used to determine the statistical difference between two observations, whether the mean difference of these two observations is significantly different from zero [https://www.statisticssolutions.com/manova-analysis-paired-sample-t-test/].

In this paper, we perform different methods on same set of data, i.e., training and testing data pairs and considering each method as a variable, paired t-test can be applied to get the significant difference between two methods. Table 6 presents the results obtained from paired t-test in terms of p-values for the battery charger

Table 3 Experimental results of T1FS and IT2FS for Box–Jenkins dataset

Testing performance in terms of MSE for Box–Jenkins dataset. (Each trial consists of average of ten runs)				
	Genetic algorithm		Memetic algorithm	
Trial	T1FS	IT2FS	T1FS	IT2FS
1	0.4871	0.3047	0.3805	0.2121
2	0.4251	0.2968	0.3986	0.1968
3	0.5566	0.2758	0.3965	0.2066
4	0.4636	0.2968	0.4025	0.2689
5	0.4958	0.3125	0.3569	0.2265
6	0.4269	0.2478	0.3756	0.2546
7	0.5218	0.2648	0.4523	0.2014
8	0.4536	0.2894	0.4025	0.2412
9	0.4739	0.3222	0.4006	0.2212
10	0.4958	0.3025	0.3865	0.2254

Table 4 Experimental results of T1FS and IT2FS for Ni–Cd battery charger dataset in terms of average, best, and worst case MSE

	Genetic algorithm		Memetic algorithm	
	T1FS	IT2FS	T1FS	IT2FS
Average case	0.49525	0.331	0.44575	0.25652
Best case	0.4473	0.2889	0.3987	0.2123
Worst case	0.5964	0.3932	0.5126	0.3012

Table 5 Experimental results of T1FS and IT2FS for Box–Jenkins dataset in terms of average, best, and worst case MSE

	Genetic algorithm		Memetic algorithm	
	T1FS	IT2FS	T1FS	IT2FS
Average case	0.48002	0.29133	0.39525	0.22547
Best case	0.4251	0.2478	0.3569	0.1968
Worst case	0.5566	0.3222	0.4523	0.2689

Table 6 *P*-values for the various results obtained for Ni–Cd battery charger dataset from paired *t*-test

Algorithm	T1FS/IT2FS	T1FS/T1FS	IT2FS/IT2FS
GA	3.4857E−08		
MA	1.0311E−07		
GA/MA		0.0002932	1.895E−05

Table 7 *P*-values for the various results obtained for Box–Jenkins dataset from paired *t*-test

Algorithm	T1FS/IT2FS	T1FS/T1FS	IT2FS/IT2FS
GA	2.15457E−07		
MA	1.11365E−07		
GA/MA		6.51186E−06	9.6285E−05

dataset. This table shows p-value between T1FS and IT2FS with GAs and MAs is 3.4857E−08 and 1.0311E−07, respectively, which is less than 0.05, and explains the significant difference between two methods. On the other hand if we compare the efficiency of GAs and MAs for same systems, that is, for T1FS and IT2FS, then *p*-values are 0.0002932 and 1.895E−05, respectively, which is less than 0.05 and indicates the significant difference between two algorithms. Table 7 gives the *p*-values for Box–Jenkins gas furnace dataset. The p-values for T1FS versus IT2FS with GAs and MAs are 2.15457E−07 and 1.11365E−07, respectively, which is very less than 0.05 and indicates the significant difference between T1FS and IT2FS for this experiment. Also the p-values for T1FS and IT2FS with GAs and MAs are 6.51186E−06 and 9.6285E−05, respectively, which shows the significant difference between two algorithms. After analyzing the results in terms of average, best and worst case MSE from Tables 4 and 5, and in terms of statistical analysis, it is clear that IT2FS performs better than T1FS for noisy data and MAs gives better efficiency than GAs.

Experiments on Other Two (Classification) Datasets

The two datasets taken from UCI repository of machine learning.are very common benchmark datasets in the field of classification. The first is iris dataset (https://archive.ics.uci.edu/ml/datasets/iris) consisting 150 patterns of three species: iris setosa, iris versicolor, and iris virginica. Species are labeled as 1, 2, and 3 with sepal length, sepal width, petal length and petal width as attributed, respectively.

Therefore, a pattern is represented as $p_k = [x_{k1}, x_{k2}, x_{k3}, x_{k4}, c_k], c_k \in [1, 2, 3], k = 1, 2, 3, \ldots, 150$, where p_k is a pattern with $x_{k1}, \ldots x_{k4}$ are sepal length, sepal width, petal length, and petal width, respectively.

Following classification rule was used to perform classification with output $z(k)$.

$$c_k = \begin{cases} 1 & \text{if} \quad z_k < 1.5 \\ 2 & \text{if} \quad 1.5 \le z_k < 2.5 \\ 3 & \text{if} \quad 2.5 \le z_k \end{cases}$$

The second dataset used is wine data classification (https://archive.ics.uci.edu/ml/datasets/wine), which employs 13 continuous variables derived from a chemical examination of wines from the same region but belonging to distinct sorts of groupings. Alcohol, malic acid, ash, ash alkalinity, magnesium, total phenols, flavanoids, non-flavanoids, phenols, proanthocyaninsm color intensity, hue, OD280/OD315 of diluted wines, and proline are the 13 continuous variables. The characteristics of ash, magnesium, total phenols, nonflavanoids phenols, proanthocyaninsm color intensity,

Table 8 Performance measure of IRIS dataset for T1FS and IT2FS with GAs and MAs

Algorithm	T1FS		IT2FS	
	Classification rate (%)	No. of misclassification	Classification rate (%)	No. of misclassification
GAs	93	11	97	5
MAs	94	9	98	3

Table 9 Performance measure of wine dataset for T1FS and IT2FS with GAs and MAs

Algorithm	T1FS		IT2FS	
	Classification rate (%)	No. of misclassification	Classification rate (%)	No. of misclassification
GAs	88.76	20	93.25	12
MAs	91.57	15	95.50	8

and hue are found to be quite consistent throughout the three classes. Hence, rests of the features are considered for modeling by ignoring these parameters.

Results and Discussions

Table 8 shows the results of most common benchmark problems of classification, i.e., iris dataset with GAs. After introducing the noise in the data, the T1FS and IT2FS are developed with GAs. The classification rate for iris dataset observed with GAs is 93% and 97%, whereas with MAs is 94% and 98% for T1FS and IT2FS, respectively. The results for wine datasets are presented in Table 9. For this dataset, the classification rate with GAs is 88.76% and 93.25%, whereas with MAs is 91.57% and 95.50% for T1FS and IT2FS, respectively. Both the results again indicate toward the better efficiency of IT2FS over T1FS and of MAs over GAs.

Further, Figs. 8 and 9 represent the convergence rate of T1FS and IT2FS for various datasets for genetic algorithms and memetic algorithms, respectively, representing the fast convergence of memetic algorithm over genetic algorithm.

6 Conclusions

This study describes the architecture for evolving interval type-2 fuzzy systems, as well as the identification of IT2FS and T1FS using memetic and genetic algorithms (GAs). After introducing noise using MATLAB's awgn() function, the proposed approach for fuzzy modeling was effectively applied to multiple datasets (a quick Ni–Cd battery charger, Box and Jenkins' gas furnace data, the iris data classification problem, and the wine data classification problem). For the studies, 30% of the data was used as a training dataset, while the remainder was used as testing datasets. The findings of the paired t-test on the experimental data show that interval type-2 fuzzy

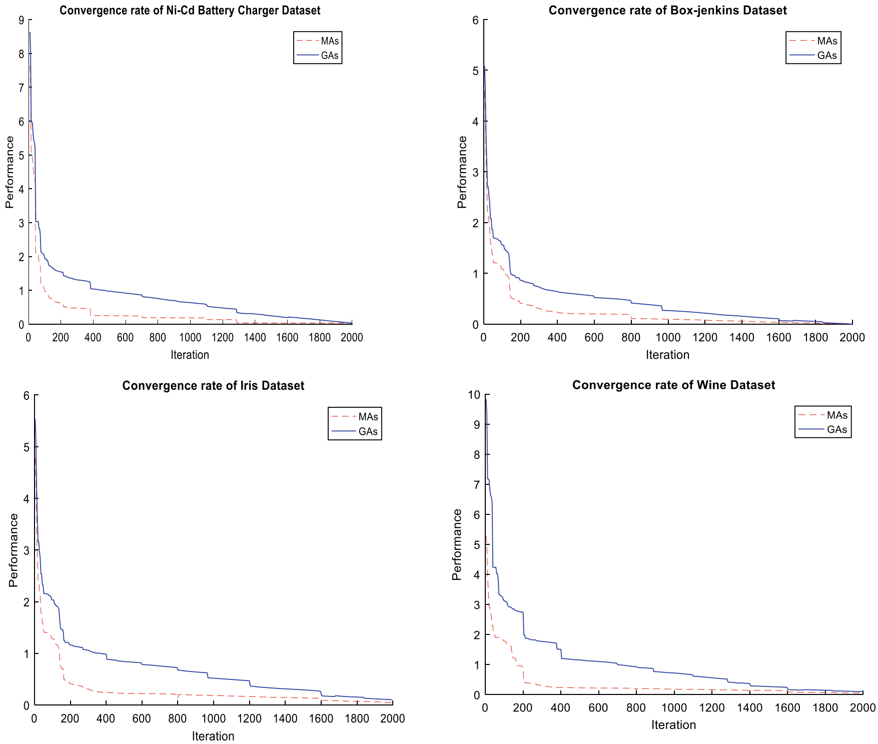


Fig. 8 Convergence rate of various datasets for T1FS with GAs and MAs

systems (IT2FS) are better at dealing with noisy data than type-1 fuzzy systems (T1FS). The research also compares the effectiveness of genetic algorithms (GAs) and memetic algorithms (MAs) in identifying fuzzy models of similar complexity for interval type-2 fuzzy systems (IT2FS) and type-1 fuzzy systems (T1FS), highlighting MAs' outstanding efficiency. The suggested framework can be expanded in the future to increase the system's versatility in terms of the number of rules, the number of membership functions for each variable, and their kind.

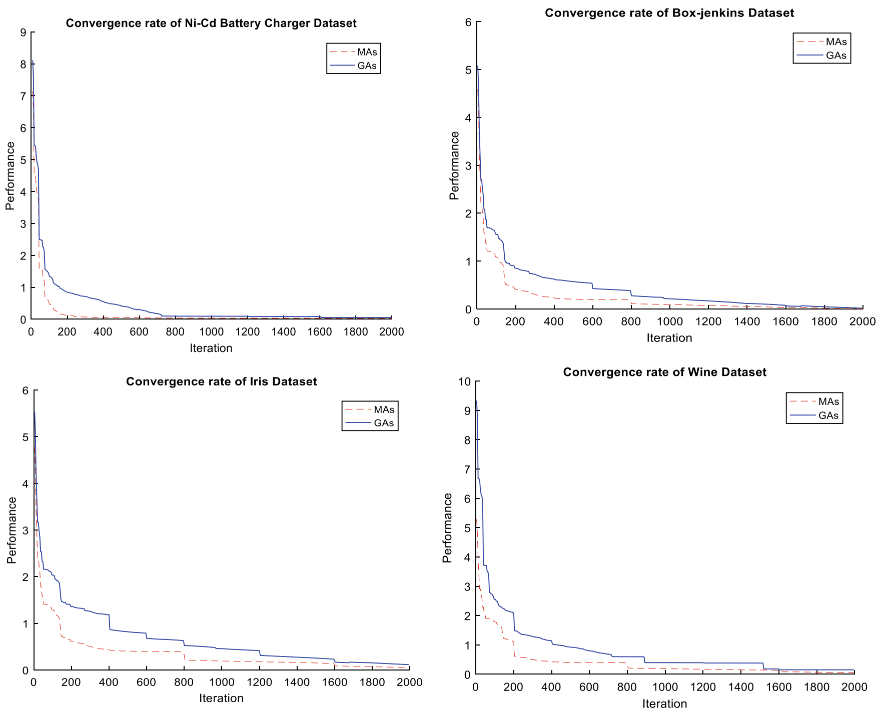


Fig. 9 Convergence rate of various datasets for IT2FS with GAs and MAs

References

1. Abed HY, Humod AT, Humaidi AJ (2020) Type 1 versus type 2 fuzzy logic speed controllers for brushless dc motors. Int J Electr Comput Eng 10(1):265
2. AbuBaker A, Ghadi Y (2020) Mobile robot controller using novel hybrid system. Int J Electr Comput Eng 2088–8708:10
3. Acampora G, D'Alterio P, Vitiello A (2018) Learning Type-2 fuzzy rule-based systems through memetic algorithms. In: 2018 IEEE international conference on fuzzy systems (FUZZ-IEEE). IEEE, pp 1–7
4. Alfi A, Fateh MM (2011) Intelligent identification and control using improved fuzzy particle swarm optimization. Expert Syst Appl 38(10):12312–12317
5. Ali F, Islam SR, Kwak D, Khan P, Ullah N, Yoo SJ, Kwak KS (2018) Type-2 fuzzy ontology–aided recommendation systems for IoT–based healthcare. Comput Commun 119:138–155
6. Araujo H, Xiao B, Liu C, Zhao Y, Lam HK (2014) Design of type-1 and interval type-2 fuzzy PID control for anesthesia using genetic algorithms. J Intell Learn Syst Appl 6(02):70
7. Baccar N, Bouallegue R (2016) Interval type 2 fuzzy localization for wireless sensor networks. EURASIP J Adv Signal Process 2016(1):1–13
8. Bansal S, Wadhawan S (2021) A hybrid of sine cosine and particle swarm optimization (HSPS) for solving heterogeneous fixed fleet vehicle routing problem. Int J Appl Metaheuristic Comput 12(1):41–65. https://doi.org/10.4018/IJAMC.2021010103
9. Bououden S, Chadli M, Allouani F, Filali S (2013) A new approach for fuzzy predictive adaptive controller design using particle swarm optimization algorithm. Int J Innov Comput Inf Control 9(9):3741–3758

10. Castillo O, Amador-Angulo L, Castro JR, Garcia-Valdez M (2016) A comparative study of type-1 fuzzy logic systems, interval type-2 fuzzy logic systems and generalized type-2 fuzzy logic systems in control problems. Inf Sci 354:257–274
11. Castillo O, Castro JR, Melin P, Rodriguez-Diaz A (2014) Application of interval type-2 fuzzy neural networks in non-linear identification and time series prediction. Soft Comput 18(6):1213–1224
12. Castillo O, Melin P (2008) Intelligent systems with interval type-2 fuzzy logic. Int J Innov Comput Inf Control 4(4):771–783
13. Castillo O, Melin P (2007) Comparison of hybrid intelligent systems, neural networks and interval type-2 fuzzy logic for time series prediction. In: 2007 international joint conference on neural networks. IEEE, pp 3086–3091
14. Cuevas-Martínez JC, Yuste-Delgado AJ, Triviño-Cabrera A (2017) Cluster head enhanced election type-2 fuzzy algorithm for wireless sensor networks. IEEE Commun Letters 21(9):2069–2072
15. Ekong U, Lam HK, Xiao B, Ouyang G, Liu H, Chan KY, Ling SH (2016) Classification of epilepsy seizure phase using interval type-2 fuzzy support vector machines. Neurocomputing 199:66–76
16. Gaxiola F, Melin P, Valdez F, Castro JR, Castillo O (2016) Optimization of type-2 fuzzy weights in backpropagation learning for neural networks using GAs and PSO. Appl Soft Comput 38:860–871
17. Hidalgo D, Melin P, Castillo O (2012) An optimization method for designing type-2 fuzzy inference systems based on the footprint of uncertainty using genetic algorithms. Expert Syst Appl 39(4):4590–4598
18. Hsu CH, Juang CF (2012) Evolutionary robot wall-following control using type-2 fuzzy controller with species-DE-activated continuous ACO. IEEE Trans Fuzzy Syst 21(1):100–112
19. https://doi.org/10.1007/978-981-16-6605-6_1
20. https://doi.org/10.1007/978-981-16-6605-6_1
21. Hwang C, Rhee FCH (2007) Uncertain fuzzy clustering: interval type-2 fuzzy approach to $ c $-means. IEEE Trans Fuzzy Syst 15(1):107–120
22. Karnik NN, Mendel JM, Liang Q (1999) Type-2 fuzzy logic systems. IEEE Trans Fuzzy Syst 7(6):643–658
23. Khosla A, Kumar S, Ghosh KR (2007) A comparison of computational efforts between particle swarm optimization and genetic algorithm for identification of fuzzy models. In: NAFIPS 2007 annual meeting of the north american fuzzy information processing society. IEEE, pp 245–250
24. Klir G, Yuan B (1995) Fuzzy sets and fuzzy logic, vol 4. Prentice Hall, New Jersey
25. Krasnogor N, Smith J (2005) A tutorial for competent memetic algorithms: model, taxonomy, and design issues. IEEE Trans Evol Comput 9(5):474–488
26. Kumbasar T, Hagras H (2014) Big bang-big crunch optimization based interval type-2 fuzzy PID cascade controller design strategy. Inf Sci 282:277–295
27. Le TL, Huynh TT, Lin LY, Lin CM, Chao F (2019) A K-means interval type-2 fuzzy neural network for medical diagnosis. Int J Fuzzy Syst 21(7):2258–2269
28. Lee CH, Chang FY, Lin CM (2013) An efficient interval type-2 fuzzy CMAC for chaos time-series prediction and synchronization. IEEE Trans Cybern 44(3):329–341
29. Li H, Sun X, Wu L, Lam HK (2015) State and output feedback control of interval type-2 fuzzy systems with mismatched membership functions. IEEE Trans on Fuzzy Syst 23(6):1943–1957
30. Li H, Wang J, Wu L, Lam HK, Gao Y (2017) Optimal guaranteed cost sliding-mode control of interval type-2 fuzzy time-delay systems. IEEE Trans Fuzzy Syst 26(1):246–257
31. Liang Q, Mendel JM (2000) Interval type-2 fuzzy logic systems: theory and design. IEEE Trans Fuzzy Syst 8(5):535–550
32. Maldonado Y, Castillo O, Melin P (2013) Particle swarm optimization of interval type-2 fuzzy systems for FPGA applications. Appl Soft Comput 13(1):496–508
33. Martínez-Soto R, Castillo O, Aguilar LT, Rodriguez A (2015) A hybrid optimization method with PSO and GA to automatically design type-1 and type-2 fuzzy logic controllers. Int J Mach Learn Cybern 6(2):175–196

34. Mendel JM, John RB (2002) Type-2 fuzzy sets made simple. IEEE Trans Fuzzy Syst 10(2):117–127
35. Mendel JM, John RI, Liu F (2006) Interval type-2 fuzzy logic systems made simple. IEEE Trans Fuzzy Syst 14(6):808–821
36. Neri F, Cotta C, Moscato P eds (2011) Handbook of memetic algorithms (vol 379). Springer
37. Nguyen HT, Sugeno M eds (2012) Fuzzy systems: modeling and control (vol 2). Springer Sci Bus Media
38. Oztaysi B (2015) A group decision making approach using interval type-2 fuzzy AHP for enterprise information systems project selection. J Multiple-Valued Logic Soft Comput 24(5)
39. Rubio E, Castillo O, Valdez F, Melin P, Gonzalez CI, Martinez G (2017) An extension of the fuzzy possibilistic clustering algorithm using type-2 fuzzy logic techniques. Adv Fuzzy Syst
40. Sanchez MA, Castro JR, Ocegueda-Miramontes V, Cervantes L (2017) Hybrid learning for general type-2 TSK fuzzy logic systems. Algorithms 10(3):99
41. Sharma AK, Mittal SK (2020) Cryptographic keyed hash function: PARAŚU-256. J Comput Theor Nanosci 17(11):5072–5084. https://doi.org/10.1166/jctn.2020.9343
42. Shukla PK, Tripathi SP (2014) A new approach for tuning interval type-2 fuzzy knowledge bases using genetic algorithms. J Uncertainty Anal Appl 2(1):4
43. Sugeno M, Yasukawa T (1993) A fuzzy-logic-based approach to qualitative modeling. IEEE Trans Fuzzy Syst 1(1):7–31
44. Wadhawan S, Goel G, Kaushik S (2013) Data driven fuzzy modelling for sugeno and mamdani type fuzzy model using memetic algorithm. Int J Inf Technol Comput Sci 5(8):24–37
45. Wadhawan S, Kumar G, Bhatnagar V (2019) Analysis of different evolutionary techniques on fuzzy rule base generation. J Comput Theor Nanosci 16(9):4008–4014. https://doi.org/10.1166/jctn.2019.8286
46. Wang W, Liu X, Qin Y (2012) Multi-attribute group decision making models under interval type-2 fuzzy environment. Knowl-Based Syst 30:121–128
47. Yao B, Hagras H, Alghazzawi D, Alhaddad MJ (2016) A big bang–big crunch type-2 fuzzy logic system for machine-vision-based event detection and summarization in real-world ambient-assisted living. IEEE Trans on Fuzzy Syst 24(6):1307–1319
48. Yeh CY, Jeng WHR, Lee SJ (2011) Data-based system modeling using a type-2 fuzzy neural network with a hybrid learning algorithm. IEEE Trans Neural Networks 22(12):2296–2309
49. Yesil E (2014) Interval type-2 fuzzy PID load frequency controller using big bang-big crunch optimization. Appl Soft Comput 15:100–112
50. Zhang T, Ma F, Yue D, Peng C, O'Hare GM (2019) Interval Type-2 fuzzy local enhancement based rough k-means clustering considering imbalanced clusters. IEEE Trans Fuzzy Syst
51. Zhang QY, Sun ZM, Zhang F (2014) A clustering routing protocol for wireless sensor networks based on type-2 fuzzy logic and ACO. In: 2014 IEEE international conference on fuzzy systems (FUZZ-IEEE). IEEE, pp 1060–1067

Secure Environment Establishment for Multipath Routing

Saju P. John, Serin V. Simpson, and P. S. Niveditha

Abstract There are a lot of challenges for mobile ad hoc networks (MANET) in the present scenario concerning certificate revocation. Suppose if there is no dynamic access to the central authority, then the certificate revocation of the malicious node is very much crucial. The spoofing of certificates by the intruders will create more threat to the secure communication system. In this paper, we propose to develop a secure multipath Optimized Link State Routing (OLSR) mechanism integrated with certificate revocation and trusted route re-computation mechanisms for MANETs, which helps to overcome these issues. According to the trust value, each node assesses the behavior of its neighbors. The proposed certificate revocation and the route re-computation mechanism minimize the overhead in multipath OLSR. As per the simulation results, the proposed approach could outperform the existing approaches in detecting the malicious nodes.

Keywords Certificate revocation · Trust route re-computation · Network resilience · MANET · OLSR

S. P. John (✉)
Department of Computer Science & Enginering, Jyothi Engineering College, Cheruthuruthy, Thrissur, India
e-mail: sajupjohn@jecc.ac.in

S. V. Simpson
Department of Computer Science & Engineering, SCMS School of Engineering and Technology, Ernakulam, Kerala, India
e-mail: serin@scmsgroup.org

P. S. Niveditha
APJ Abdul Kalam Technological University, Thiruvananthapuram, Kerala, India

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_2

1 Introduction

1.1 Mobile Ad Hoc Networks

Wireless networks that support multi-hop interactions and are self-configuring and self-organizing are known as mobile ad hoc networks (MANETs). Depending on the needs of the network, this definition allows the creation, combination, or division of the network into multiple networks. In addition to wireless cellular networks, ad hoc networks can also be set up without a base station. Routes between the end users in such a network have multi-hop wireless links. Additionally, ad hoc networks have the ability to move independently [1].

1.2 Attacks in MANET

The likelihood of security attacks is higher in MANETs than in wired networks. In addition to the lack of certification authorities, centralized monitoring, and restricted security of individual nodes, other factors such as uneven network performance make security more difficult. Wireless networks are vulnerable to attack from all directions. In this case, each node must be ready to handle attacks either directly or indirectly. MANETs are prone to passive and active attacks, especially attacks that originate from a malicious node inside the network, which can cause big damage and are difficult to identify [2, 3].

1.3 Certificate Chaining Approach

Two nodes may exchange public keys if they wish to exchange secure communications using the technique of verifying and signing every packet sent across the network. As a part of this method, encryption keys are signed by each hop, and then the next hop verifies the signature. It is called certificate chaining. This approach has the advantage that the public keys can be transmitted securely to the destination.

1.4 Need for Certificate Revocation

Mobile ad hoc networks (MANETs) have greatly increased in popularity in recent years. There are no fixed infrastructures in MANETs, and nodes can freely join and leave, because they are highly flexible. As a result, they are vulnerable to attacks from malicious nodes. MANET is susceptible to attacks by its very nature. There are several challenges in MANET security, including confidentiality, integrity, authenticity, availability, and reliability. One of the widely used authentication mechanisms in digital networks is certificate revocation [4].

1.5 Advantages

When the node's expiry time (ET) elapses, the node broadcasts a renewal request packet (RWREQ) to its neighbors in the certificate revocation technique, thereby reducing the attacks of malicious nodes in the network [5, 6].

1.6 Drawback

The certificate revocation list should be updated periodically. All the updates need to be circulated in the network without delay. This will enhance the complexity of the same.

1.7 Problem Identification and Contribution

As we can see from the existing works, there is a need for good self-certified key generation mechanisms and a secure certificate exchange model. It is challenging to revoke certificates in mobile ad hoc networks (MANETs) where there is no online access to trusted authorities. As part of this paper, we propose to design a certificate revocation mechanism that is integrated with the routing protocol. The following are the four phases of our approach.

Phase 1—Trust management mechanism.
Phase 2—Certificate exchange technique.
Phase 3—Certificate revocation.
Phase 4—Trusted route re-computation.

2 Related Works

Gurpreet et al. [1] proposed a novel system to extend the multipath routing algorithm into the wireless communication scenarios. Swarm optimization-based routing technique is one of the widely used routing techniques. The authors have incorporated the swarm optimization technique in the proposed work for the efficient routing. Along with swarm-based routing, the authors have also used the multipath ant colony technique in this paper. The authors have also given a comparison table of the various routing techniques, which will be useful for selecting the best routing protocol for a particular application.

Reddy et al. [2] proposed an efficient technique to find the secure routes utilizing the key exchange approach. The technique is based on the asymmetric key authentication. The proposed routing technique provides secure routing over all kinds of security attacks. The advantage of the proposed system is that the authors have tried to develop a routing technique which considers both the quality of service and the security of the network together, and they have considered different kinds of attacks in a single paper. Generally, each paper consider a single security issue.

Singh et al. [4] proposed a novel technique named T-DelpHI to detect the presence of wormhole attacks. Wormhole attacks are generally very difficult to identify. The authors have utilized the route reply time to evaluate the presence of wormhole attack. A threshold value of route reply time is assigned for every node. This threshold route reply time is compared with the actual time taken for the node to get the route reply, and based on these values, the presence of wormhole in the network is detected.

Singh et al. [5] proposed a protocol link based on the expiration time. It is a time-based routing protocol. The authors have used the method of calculating different link expiration time, basically maximum, minimum, and the average. The expiration time was calculated using the greedy algorithm. The expiration time for each node is calculated, and it was updated periodically. This helps to evaluate the authenticity of the transmitted data packet. The packet number purely depends on the bandwidth. The main advantage of the proposed work is its low complexity.

Liu et al. [7] proposed a novel routing technique to deal with the anonymous communications. In mobile ad hoc networks, the nodes are mobile, and due to this reason, the nodes will have to communicate with unfamiliar nodes. The security risk factor is very high in such type of communications. There are many anonymous routing protocols but all of them have limitations in case of detecting fake routing packets. In this paper, the authors have proposed authenticated anonymous secure routing (AASR) for MANETs, which effectively resists the attack in anonymous communication. The main two techniques described in this paper are group signature technique and key encryption method.

Sapna et al. [8] shared a detailed review on various routing protocols of mobile ad hoc networks. When comparing the performance of different protocols, a variety of parameters, such as throughput, packet delivery ratio, delay, and jitter, are considered. These authors have compared three major routing protocols in order to develop this paper: AODV, OLSR, and DSDV. Using this review paper, you can easily select the protocols based on the applications.

3 Proposed Solution

3.1 Overview

Our paper proposes an improved protocol for link state routing in MANET that integrates certificate revocation. Every node in this system monitors the behavior of its neighbor node and accumulates the trust value for each node monitored. The node marks a neighbor as malicious if the trust value of that neighbor is below the minimum threshold. A certificate revocation list is created for the detected malicious node. Once the malicious behavior has been detected, the node will notify the source. In its routing table, the source then records the path number and node details of the malicious node that was detected. In order to protect against the malicious nodes in the path, the source node discards them and bypasses the data packet through other nodes in another selected path toward D using multipath technique, while implementing the certificate revocation process. In order to defend against mobility, a recomputed route has been used.

3.2 Trust Management Mechanism

%F (i, j) and %E (i, j) are computed by the node Ni. %F (i, j) is defined as the percentage of packets initiated from Ni which were forwarded by n_j over the total number of packets offered to n_j. %E (i, j) is defined as the percentage of packets that were expired over the total number of packets offered to node j. The recent satisfaction index (RSI) is calculated based on the number of packets successfully reached at the destination. The expiry rate of the packets defines the index. Based on the eigen vector centrality, the trust is calculated for each node [9].

3.3 Detection of Misbehaving Nodes

Source X and destination Y are respective sources defining the minimum threshold for trust as T(M). Throughout the transmission range of a node, Ni monitors the calculated trust value T(M) by its neighboring nodes. Nodes monitored by Ni send their trust values to their neighbors. The node that identifies the malicious node after the information exchange adds the node's information to its certificate revocation list (CRL) if it discovers that the trust value of the monitored node is below T(M). If malicious behavior is detected, X receives a warning message from the node that detected it. The malicious node is recorded in the routing table of X along with the path number and node ID. A source node forwards the data packet to D by avoiding the malicious nodes on that path (multipath technique discussed in Sect. 3.3.1) and uses certificate revocation to defend itself from malicious nodes (explained in Sect. 3.3.2).

During the data transfer, if any node is not available due to mobility, a trusted route re-computation mechanism is also employed (explained in Sect. 3.3.3).

3.3.1 Certificate Exchange Mechanism

Nodes need to verify themselves before they can access the network resources with the help of certificate exchange. In order to improve certificate exchange protocol reliability, multi-path technique has been developed. In this case, the certificate exchange mechanism relies on the OLSR protocol [9].

3.3.2 Certificate Revocation

In order to defend against malicious nodes, the source performs certificate revocation. As a first step, this process assumes the following. The CRL initially will be empty. During the communication process in a random interval, the CRL update process will happen. Then, the nodes having the below threshold trust are included in the CRL list. Some nodes may retain its trust after mobility issue; these nodes are released from the CRL list. During the path selection, the source node discards the nodes that are included in the list from the trusted path [9].

3.3.3 Trust-Based Route Recovery Mechanism

Maintaining route recovery is ensured by the MOLSR multipath route recovery protocol, which implements a trust-based system. It is possible that a particular path might become unavailable during a transmission due to mobility or a broken link after certifying the paths. Prior to forwarding a packet to the next hop, an intermediate node in MOLSR first verifies whether the neighbor node is valid. The node will use its best efforts to recompute the route and forward the packet using the new route if the neighbor node is invalid, as indicated in Fig. 1. To overcome different types of security attacks, the proposed solution enhances the existing OLSR protocol.

4 Simulation Results

4.1 Simulation Model and Parameters

A simulation was performed with network simulator (NS-2), a tool particularly popular within the ad hoc networking community. All simulations use IEEE 802.11 with a data rate of 11 Mbps for the MAC layer 250 m is the range of the transmission. Two-ray ground is the propagation model. In a 1000 $\times$ 1000 m network area, there

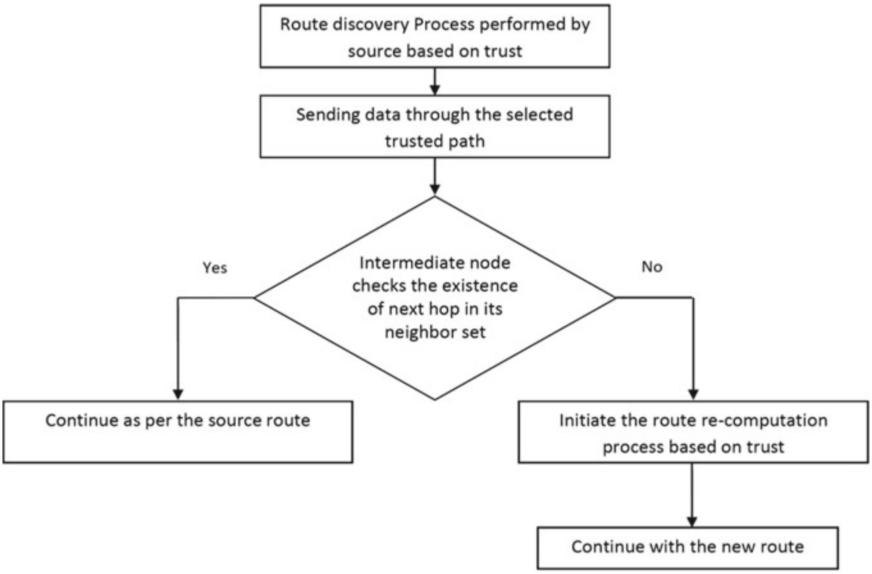


Fig. 1 Trusted route re-computation

are 100 nodes. A minimum speed of 5 m/s has been determined by our simulation network pairs of sources, and destinations are dispersed at random. To set up the pattern of connections, NS-2 constant bit-rate (CBR) traffic generator is used. All nodes are connected to a single CBR traffic destination. Over the course of 60 s, the initiation time of the sources is uniformly distributed. Load values range from 50, 100, 150, 200, to 250 Kb. Five hundred twelve bytes were also set as the certificate size. Seven connections were set up in the network. In the simulation, false certificates are sent by attacking nodes to those nodes which requested them. Attackers can use different public keys to certify a different public key for each attack. In addition, the attackers may collaborate and send certifications for the same public key that is spurious, resulting in a cooperative attack. Both isolated and collusion attacks are simulated. In the network, the percentage of attacker nodes is fixed at 10% of the total number of nodes (i.e., ten attackers). Our simulation settings and parameters are summarized in Table 1

Table 1 Simulation settings

Parameters	Values	Parameters	Values
No. of nodes	100	Traffic source	CBR
Area size	1000 × 1000	Packet size	512
Mac	802.11	Speed	5 m/s
Radio range	250 m	Traffic source	CBR
Simulation time	50 s	Load (Kb)	50,100,150,200,250

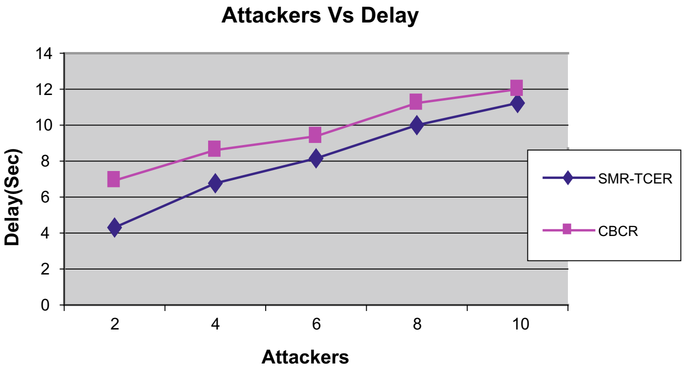


Fig. 2 Attackers versus delay

4.2 *Performance Metrics*

Secure multipath routing technology incorporated with Trusted Certificate Exchange and Revocation (SMR-TCER) is compared with cluster-based certificate revocation (CBCR) [10]. Based on these metrics, the performance of the gateway is measured: average latency, packet delivery ratio, reliability, packet drop, and detection ratio.

In Fig. 2, the average delay between the two schemes is shown as attackers are increased from 2 to 10. We can see that as the attacker increases, the delay increases linearly. The existing CBCR scheme takes 17.5% longer than our proposed SMR-TCER. According to Figs 4 and 3, respectively, the CBR data packets dropped by malicious attackers are shown. There are more data packets dropped when there are more attackers. When compared with CBCR scheme, SMR-TCER has a 28% decrease in packet losses. In Fig. 3, the packet delivery ratio has decreased as a consequence of linearly increasing packet drops. When compared to CBCR, SMR-TCER has a 14% higher packet delivery ratio. Figure 5 shows that capturing nodes was not resisted significantly. SMR-TCER has fewer compromised nodes since it has a trusted mechanism. As a result, SMR-TCER has 29% less resilience than CBCR. As shown in Fig. 6, miss detection ratio results have been calculated. When compared with CBCR, SMR-TCER has a 69% lower miss detection ratio.

5 Conclusion

This paper proposes a secure multipath OLSR technology, coupled with route re-calculation and certificate revocation in MANETs. The trust values of each node will be calculated by using the trust management mechanism, and then, the path selection process is done. The certificate revocation list includes nodes that do not meet the minimum threshold of trust. Source nodes eliminate malicious paths during path

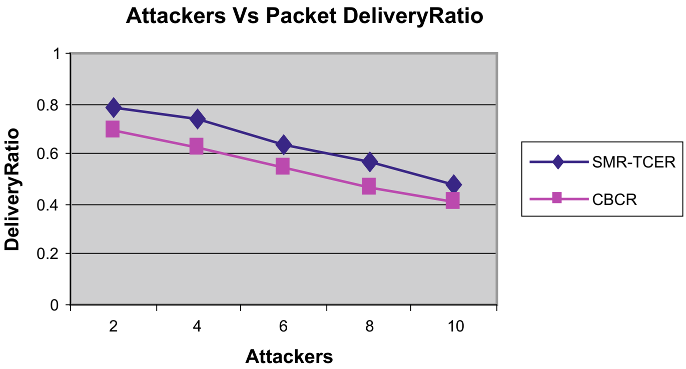


Fig. 3 Attackers versus delivery ratio

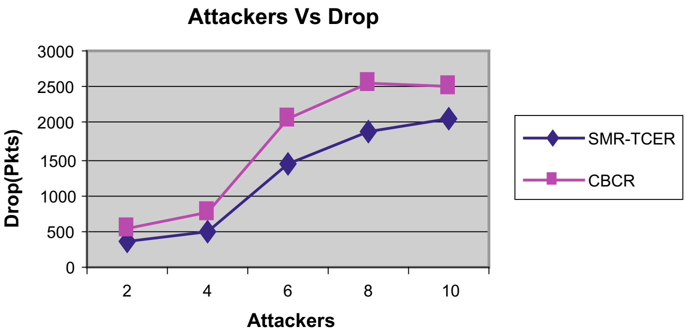


Fig. 4 Attackers versus drop

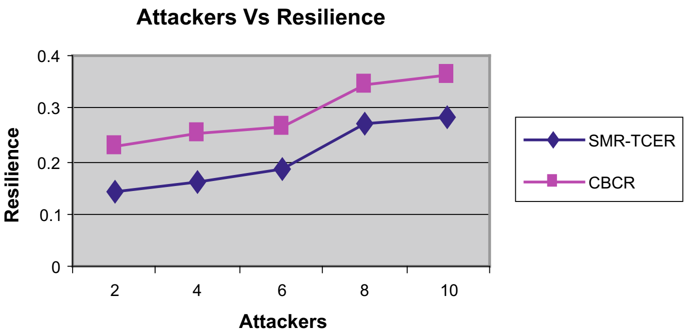


Fig. 5 Attackers versus resilience

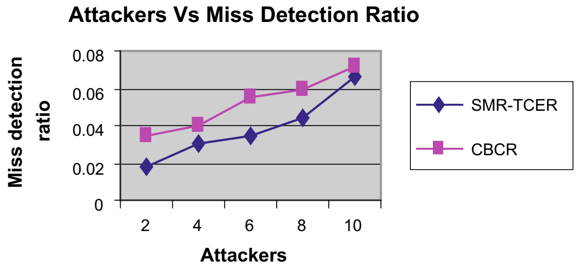


Fig. 6 Attackers versus misdetection ratio

selection and thereby protect against attackers. Due to the mobility, if a neighboring node is not available during the communication stage, then a route re-computation mechanism is employed to select a route at that time. The certificate revocation and route re-computation mechanisms greatly improve the resilience and the detection ratio of the network.

References

1. Singh G, Kaur A (2019) Exploration of multipath routing protocol for mobile ad hoc networks. Int J Sustain Agric Manage Inf 5(4)
2. Raju LR, Reddy CR (2017) A key exchange approach for proficient and secure routing in mobile adhoc networks. IJIM 11(4)
3. Simpson SV, Nagarajan G (2021) A fuzzy based co-operative blackmailing attack detection scheme for edge computing nodes in MANET-IOT environment. Future Gener Comput Syst 125:544–563
4. Singh N, Singh A (2018) An improved (T-DelPHI) for efficient and secure routing in MANETs. In: 4th international conference on computers and management (ICCM 2018)
5. Singh G, Rohil H, Rishi R, Ranga V (2019) LETSRP: a secure routing protocol for MANETs. Int J Eng Adv Technol (IJEAT) ISSN 9(1):2249–8958
6. Simpson SV, Nagarajan G An edge based trustworthy environment establishment for internet of things: an approach for smart cities. Wireless Networks, 1–17
7. Liu W, Yu M (2014) AASR: authenticated anonymous secure routing for MANETs in adversarial environments. IEEE Trans Veh Technol
8. Sapna T, Deshpande K, Ravi K (2018) Study on routing protocols for MANETs. In: 2018 international conference on computational techniques, electronics and mechanical systems (CTEMS)
9. John SP, Samuel P (2015) Self-organized key management with trusted certificate exchange in MANET. Ain Shams Eng J 6:161–170
10. Liu W, Nishiyama H, Ansari N, Kato N (2011) A study on certificate revocation in mobile ad hoc networks, IEEE

Comparative Analysis of Transfer Learning and Attention-driven Memory-based Learning for COVID-19 Fake News Detection

Anshika Choudhary and Anuja Arora

Abstract In the pandemic COVID-19 situation, the world is facing a pandemic of fake information which often stirs the public attention by attacking their emotional quotient. Scenario reached a situation where people in search of worthy information for public health and precaution, getting fake news. This unprecedented expansion of fake information has become a challenging research issue. Deliberate efforts have been attempted in this manuscript for finding a solution to this COVID-19 fake news detection problem with the help of deep learning models. Two deep learning models—BERT, a transfer learning model, and attention-based bi-directional long short-term memory (LSTM), a memory-based model, have been applied in order to get accurate fake news classification outcomes. A comparative outcome of both models is presented which shows BERT outperforms and gives excellent results in comparison to the attention-based bi-directional LSTM model. The achieved training accuracy by BERT is 86% which is much higher than the accuracy achieved by attention-based Bi-LSTM. BERT precision, recall, and F-score are 0.82, 0.79, and 0.80, respectively, which shows that BERT can detect COVID-19 fake news better than the attention-based Bi-LSTM model.

Keywords Transfer learning · BERT · LSTM · Attention

1 Introduction

The recent rapid advancement of technology in online social webs such as Twitter, Instagram, and Facebook has driven a great incline in the propagation of fake news. Recently, the COVID-19 virus, which was primarily detected in China (Wuhan) in December 2019, has declared a public health emergency of international concern

A. Choudhary (✉) · A. Arora
Department of Computer Science and Engineering and Information Technology, Jaypee Institute of Information Technology, Noida, India
e-mail: anshika.ch412@gmail.com

A. Arora
e-mail: anuja.arora29@gmail.com

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_3

on January 30, 2020, by the World Health Organization (WHO). The whole world was gripped by a pandemic named coronavirus. A declared contagious disease has covered more than 75 million confirmed cases and 1.7 million deaths across the world till December 01, 2020, to the WHO [1]. During the pandemic, the fake news puts lives at risk. Unfortunately, the misleading news about COVID-19 has encouraged the growth of disease and chaos among individuals, which caused serious social disruptions. The WHO has stated the dissemination of coronavirus named COVID-19 as an "infodemic".[1] An infodemic is an overabundance of information flowing over the Internet that causes confusion and undermines the trustworthy sources and public health response when they need to protect their health. The prevalent wrong information is disrupting social order. Recently, for example, 77 mobile towers are burnt in the UK last year (2020) which presents a conspiracy theory being spread online blaming radiation from 5G towers for the ongoing pandemic.[2] Claiming that it weakens the human immune system, thus helping the coronavirus spread faster. Thus, mitigating the spread of COVID-19 associated fake news has become a critical problem which is gaining notable attention from the government to global health organizations (WHO), online social networking sites, and news organizations (BBC, CNN, and New York Times). Any single false information that gains huge traction can disprove the significance of multiple verified facts/realities. However, demystifying COVID-19-related false information exhibits its own set of unique challenges and became a task of utmost importance to spot the spreading. Due to house arrest during the COVID-19 pandemic, Twitter witnessed an average of 30% rise in daily usage [2]. During the COVID-19 period, isolated users from the external world were directed to social media platforms for any updates related to the COVID-19 pandemic. Quite often, people are misinformed when unverified news is shared irrespective of intentions. With the increase in the consumption of social media platforms, fake news tends to increase rapidly. Therefore, to aid the complexity of the issue related to the authenticity of the COVID-19 content, this paper tries to fix the challenge and compares the performance of two learning paradigms specifically for the detection and classification of fake news for COVID-19. These two paradigms are bi-directional encoder representations from transformers (BERT), which is a pre-trained model, and attention-based bi-directional LSTM. Fundamentally, both are attention-driven models, but this work has been done in order to present the performance of BERT over hybrid attention-integrated Bi-LSTM. The research contribution of the presented work is as follows:

- Explored the performance of transfer learning model BERT and attention-based bi-directional LSTM for COVID-19 fake news detection.
- A comparative study of attention-driven deep learning models—BERT and attention incorporated Bi-LSTM—is presented for fake news detection and classification for COVID-19.

---

[1] https://www.who.int/health-topics/infodemic.

[2] https://www.businessinsider.com/77-phone-masts-fire-coronavirus-5g-conspiracy-theory-2020-5.

- The accuracy achieved by BERT is exceptionally outstanding on both training and test dataset.

The rest of the paper structure is organized in such a manner. Section 2 reviews the existing work on fake news detection, especially studies done on the COVID-19 news datasets using deep learning architectures. In Sect. 3, we present the learning models validated on COVID-19 fake news detection, and Sect. 4 describes the dataset that we used in experiments, showcases our experimental results, and results analysis. Finally, we conclude the paper in Sect. 5.

2 Related Work

Fake news detection has been observed in various ways, starting from recognizing the spread of false information, how it is formed, and how it disrupts the society. Up to today, so many approaches, models, and architectures have been developed to detect misinformation [3]. Earlier various approaches were employed to detect the online social media content, including feature extraction adaption [4, 5], data orientation, and model/architecture adaption [5]. From different levels of granularity of the language structure, various textual features are extracted to represent the fake news from the aspects of words, sentences, messages, topics, and events. Much explored approaches are feature extraction which concentrates on the content and social context of the news [6]. These linguistic features are derived from the text [7], such as linguistic cues selection that includes lexical, syntactic, and topic features [4, 8, 9], which is treated as an important task in many natural language-driven understandings. In 2021, the proposed linguistic feature-based fake news detection model evoked the researcher's attention [4] toward various language-driven features that are the main cause of outcome refinement, whereas social context features include user features such as [10, 11], propagation features [12–14] and temporal features [15, 16], which are also attempted by the research community. Hence, feature selection is considered the most significant task. Heading toward this research area, immense work done is carried out using various traditional classification approaches. Castillo et al. 2011 [10] analyze the information credibility of news propagated through Twitter while projecting a number of learning classification schemes such as random forest and support vector machine (SVM) [17], decision trees [7, 10], and Bayes networks [18] for handcrafted features. These conventional traditional machine learning approaches often result in a high-dimensional representation of linguistic information, leading to the curse of dimensionality where excessive sparse matrices have to be taken into consideration. This issue further can be resolved by using word encoding representation which delivers low-dimensional distributed representations. Word encoding schemes like continuous bag-of-words (CBoW) and skip-gram [19] are commonly used to evaluate continuous vector representations from a corpus of text. Word2Vec and GloVe are the global word vectors representation [20] that are applied in association with pre-trained embeddings like bi-directional pre-training formation (BERT) that use the transformer-based encoder [21] and are measured. Moving forward, the text corpus holds a sequential time series data which has some dependencies between what words immediately follow and precede a word in each sentence. Recurrent neural networks have been extensively adopted to fix natural language processing (NLP) complications. In 2016, Ma et al. [22] proposed a framework for rumor debunking by making use of recurrent units which utilize the variances of aggregated information of each event across different time intervals. Next, the CSI-capture, score, and integrate model is composed of three modules that implement recurrent neural networks [23] to capture the temporal pattern of user activity, learning the source characteristic based on the user's behavior, and finally classifying the article as fake or not. Furthermore, researchers try with hybrid architecture using multi-domain features. In 2019, analyst [24] stemmed out the challenge on different modalities (text + visual) representations by announcing multimodal variational autoencoder. And in 2020 [25], Agarwal et al. proposed the combination of convolution neural network (CNN) and recurrent neural network-LSTM (RNN) architecture for predicting and detecting the fake news [5]. The novelty is interpreted with the observation of hyperparameter tuning values. Hence, the proposed model together (CNN and RNN) achieved better accuracy.

As the news article encompasses a large corpus, this makes the input sequence long enough. In such a scenario, the past information gets vanished and scatters focus over the sequences, which is due to a lack of explicit word alignment during the decoding process. This is the place where urgency is raised to solve this issue, and the attention mechanism turns out to be an improvement over the sequential model. Recently, researcher [26] projected a method by fusing text and visual features offering multi-model with BERT and attention. Heejung et al. [27] designed the bi-directional encoder representations from transformers model (BERT) to analyze the relationship between the title and the description of the news [28]. Especially for COVID-19, numerous misleading contents or fake news remain online on social media platforms. Many of them disseminate false information that causes panic among society, misguides people, and thus aggravates the consequences of the pandemic [29]. Glazkova et al. [30] utilized the transformer-based ensemble model and proposed a final method as of COVID-19 Twitter-BERT (CT-BERT). This work presents the results in a shared task of COVID-19 fake news detection in the English language. An ensemble of three transformer models—BERT, ALBERT, and XLNET—for fake news detection is discussed in [3, 31, 32]. Wani et al. [33] figure out various deep learning algorithms based on convolutional neural networks (CNN) as well as recurrent neural network approach such as long short-term memory (LSTM) and more advanced transformer-based BERT model on COVID-19 corpus [29, 34]. Analysts came up with a number of findings in this research community but lack in defining the parameter setting of the applied model. Recently, in 2021 [2], researchers suggested a COVID-19 Twitter fake news dataset as the first labeled dataset, along with a huge set of unlabeled tweets. Here, the authors presented a semi-supervised model named cross-SEAN and a chrome extension version of cross-SEAN, which helps in verifying the tweets as fake and genuine and can help to collect labeled data based on user feedback.

3 Learning Models Validated for COVID-19 Fake News Detection

3.1 *Bi-directional Encoder Representations from Transformers (BERT)*

The transformer is the first transduction architecture that entirely relying on self-attention mechanism to compute the representations of its input and output element without using sequence aligned order as RNNs or convolution. The motive of the transformer using attention and recurrence mechanism is to grasp the dependencies between input and output aspects. This model led to parallelization of the process, which made it possible to leverage the contemporary hardware. The transformer model architecture consists of an encoder-decoder block structure. The transformer comprises a multi-head self-attention layer and stacked on top of each other followed by a feed-forward neural network layer where the output of one block is passed as input to another stack block. The self-attention mechanism's responsibility is to look at all the surrounding words in the given input sentence to know the full context of a certain word in the input sequence.

Undoubtedly, the transformer-inspired BERT are paving the way, and all the following breakthroughs happen in the natural language processing. BERT refers to a deeply bi-directional model, which means it is designed to learn the context of the input sequence in both directions at once during the training phase. The BERT architecture builds on top of the transformer model. It consists of variants, out of variants, BERT base has been applied in the implementation process, which comprised of 12 layers as (transformer blocks) with 12 (attention heads) and 110-M parameters. BERT base does not have a decoder but follows a 12 stacks encoder. BERT expects an input representation which is constructed by summing up the combination of three embeddings such as token embedding, segment, and positional embedding. At the beginning of the first sentence, a [CLS] is added as a special classification token. This token is used as a combination of the entire input sequence representation that is considered as the final hidden state to this token and fed to the classification layer. While fine-tuning, this classification layer is the only constraint adjusted to the model according to the specific related task. To differentiate the input sentence, a special [SEP] token is injected at the end of each input sequence and by adding a segment embedding representation to every input token indicating whether it belongs to sentence A or sentence B. As depicted in Fig. 1, E is the input token embedding, [CLS] a special token as C is considered as the final hidden vector, and T_i as the final hidden vector for the ith input token. BERT holds an input token sequence, and it keeps moving up the stack by stack. At each stacked block, it is first passed through a self-attention layer and then moved to a feed-forward neural network, then continuing passing on to the next encoder. For each task, simply pad the task-specific

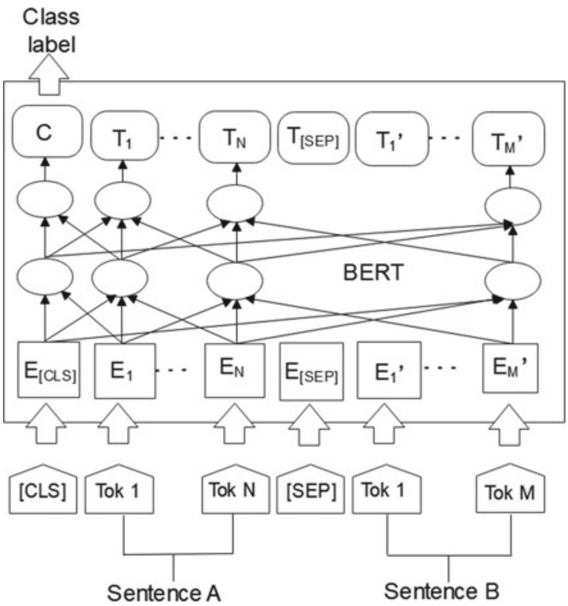


Fig. 1 Schematic diagram of bi-directional encoder representations from transformers learning (BERT) model [21]

inputs and outputs in the direction of BERT model and lastly fine-tune all the parameters/dimensions end to end. We used a twelve layers in BERT model, where a token will have 12 intermediate learning representations.

3.2 Attentional-based Bi-directional Long Short-term Memory Model

The attention mechanism turns out to be an improvement over the previous sequential model. It was initially designed in the context of neural machine translation. Additionally, the proposed architecture can exhibit the ability to indicate a selective focus on subregions of inputted fake news and store the extra information in memory vectors. The purpose of the attention mechanism is to concentrate on certain important parts of the input sequence during the training so that the model pays close attention to the information. The attention bi-directional LSTM model contains five layers. The bottom layer is an input layer followed by an embedding matrix layer in which an embedding matrix is used for all the news taken into consideration. The third layer is the bi-directional LSTM layer where $h_t^{\rightarrow}$ and $h_t^{\leftarrow}$ present the forward and backward layer, respectively. The fourth layer is the attention layer where based on the above-mentioned process, attention is computed in a continuous manner in both forward and backward directions. Finally, the last layer symbol means the summation of all-important context vectors of specific news and classifying it (Fig. 2).

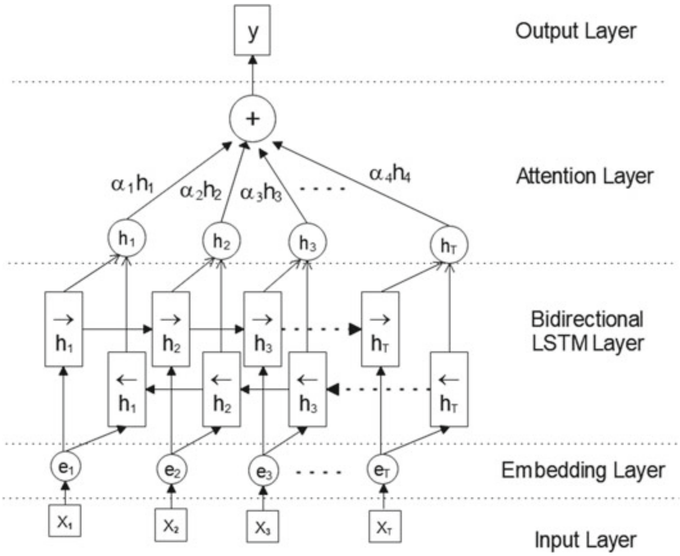


Fig. 2 Schematic diagram of attention-based long short-term memory model (LSTM)

4 Experimental Setup and Model Performance Evaluation

4.1 Dataset Details

COVID-19 fake news dataset[3] has been used to validate the outcome of applied deep learning models. Metadata information of taken dataset is news id, news title, news text, news subcategory, and label. The subcategories are divided into three different categories such as false news, partially false news, and true news. Both the "false news" and "partially false news" are labeled as "0", and "true news" is labeled as 1 for the classification task. So, the dataset contains binary labels of news, i.e., true or fake news of COVID-19. The dataset collection period for COVID-19 news starts from December 2019 up till July 2020 [35]. This dataset was collected by Webhose.io which is labeled manually. A screenshot of the dataset for readers' understanding is displayed in Fig. 3.

News title and news text concatenated for fake news detection experiment. The dataset covers a total of 3119 news, out of which the fake news count is 1058 and the true news count is 2061.

[3] https://data.mendeley.com/datasets/zwfdmp5syg/1.

Unnamed: 0	title	text	subcategory	label
0	FACEBOOK DELETES MICHIGAN ANTI-LOCKDOWN GROUP ...	Facebook has shuttered a popular group for Mic...	false news	0
1	Other Viewpoints: COVID-19 is worse than the flu	We can now officially put to rest all comparis...	true	1
2	Bermuda's COVID-19 cases surpass 100	The Ministry of Health in Bermuda has confirme...	true	1
3	Purdue University says students face 'close to...	Purdue University President Mitch Daniels, the...	partially false	0
4	THE HIGH COST OF LOCKING DOWN AMERICA: "WE'VE ...	Locking down much of the country may have help...	false news	0

Fig. 3 Screenshot of COVID-19 dataset taken for research study

4.2 *Experimental Outcome*

In this section, the performance of the transfer-based learning model and attention-driven bi-directional LSTM is compared and validated using various evaluation metrics such as accuracy, precision, recall, and weighted F1-score. The training and validation average achieved accuracy is 0.86 and 0.84, respectively, using the BERT deep learning model. On the other side, attention-based LSTM achieved an average accuracy as of 0.65 and 0.70, respectively, for training and validation sets. The reported accuracies using both applied deep learning models are tabulated in Table 1.

The accuracy plot of BERT and attention-based bi-directional LSTM is shown in Fig. 4a and b which clearly depicts the performance of BERT is considerably much better than attention-driven Bi-LSTM model. BERT demonstrates noticeably improved performance with validation accuracy as of 0.84 and an F1-score as of 0.80 on the test set that is a tremendous increase in accuracy and other measures in comparison to attention-based bi-directional LSTM.

Table 1 Training and validation accuracy comparison of BERT and attention-based Bi-LSTM DL model

Technique	Training accuracy	Validation accuracy	Precision	Recall	F1-score
Bi-LSTM attention	0.65	0.70	0.35	0.50	0.41
BERT base-L-12	0.86	0.84	0.82	0.79	0.80

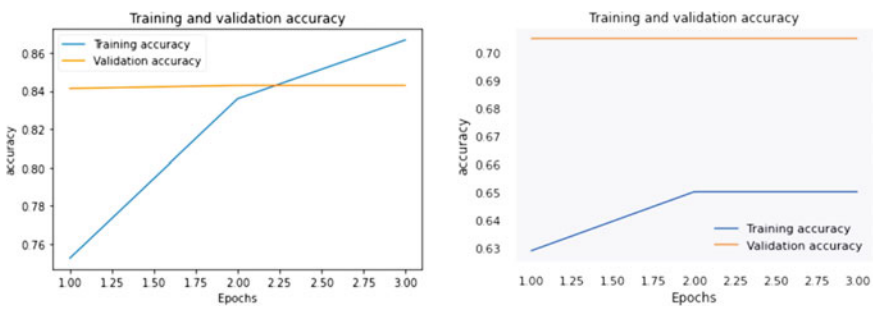


Fig. 4 Training and validation accuracy plot **a** BERT, **b** attention-based Bi-LSTM model

4.3 Discussion

In this paper, attention-based bi-directional LSTM and BERT base uncased version are adopted, and a comparative analysis is presented for the fake news detection task. From the experiment results reported in Table 1, it is depicted that the transformer-based model is considerably much better for our COVID-19 fake news detection and classification task. BERT base has the best effect, outperforming with noticeable validation accuracy as 0.84, precision as 0.82 with recall as 0.79, and a tremendous increase in F1-score as 0.80 at epoch 3. Through the analysis, it is observed that the results of the transformer-based model on our COVID-19 dataset are remarkable and thus can be used for fake news detection and classification task. BERT architecture extracts the detailed context between words and sentences, which verifies the effectiveness and feasibility of the approach adopted in this paper.

5 Conclusion and Future Scope

In this manuscript, the two deep learning approaches are presented to combat the COVID-19 pandemic. The proposed transfer-based model produces a better result than the Bi-LSTM attention mechanism depicting the best training and validation accuracy as 0.86 and 0.84. Deception of fake news is prevalent in such ongoing global infodemic COVID-19, leading to individuals accepting fatal information. Quick detection of fake news can diminish the spread of fear and confusion among the audience that claims human lives. Therefore, a methodology is projected to examine the credibility of information on social media pertaining to the global crisis. The proposed model limits the deception of fake news particularly in a precipitous situation like the COVID-19 global infodemic. The leveraging state-of-the-art advanced transformer-based NLP model can address the spread of fake news. Future research work extension can include varied extensive features such as visual-feature driven and incorporate semantic information from images plus textual information, which are readily available with the news.

References

1. Patwa P, Bhardwaj M, Guptha V, Kumari G, Sharma S, Pykl S, Das A, Ekbal A, Akhtar MS, Chakraborty T (2021) Overview of constraint 2021 shared tasks: detecting English covid-19 fake news and Hindi hostile posts. In: International workshop on combating on line hostile posts in regional languages during emergency situation. Springer, Cham, pp 42–53
2. Paka WS, Bansal R, Kaushik A, Sengupta S, Chakraborty T (2021) Cross-SEAN: a cross-stitch semi-supervised neural attention model for COVID-19 fake news detection. Appl Soft Comput 107:107393
3. Jindal R, Dahiya D, Sinha D, Garg A (2022) A study of machine learning techniques for fake news detection and suggestion of an ensemble model. In: International conference on innovative

computing and communications. Springer, Singapore, pp 627–637
4. Choudhary A, Arora A (2021) Linguistic feature based learning model for fake news detection and classification. Expert Syst Appl 169:114171
5. Singla M, Dutta M (2022) Deep classifier for news text classification using topic modeling approach. In: International conference on innovative computing and communications (pp 139–147). Springer, Singapore
6. Cao J, Guo J, Li X, Jin Z, Guo H, Li J (2018) Automatic rumor detection on microblogs: a survey. arXiv preprint arXiv:1807.03505.
7. Gravanis G, Vakali A, Diamantaras K, Karadais P (2019) Behind the cues: a benchmarking study for fake news detection. Expert Syst Appl 128:201–213
8. Su T, Macdonald C, Ounis I (2019) Ensembles of recurrent networks for classifying the relationship of fake news titles. In: Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval, pp 893–896
9. Wu K, Yang S, Zhu KQ (2015) False rumors detection on sina weibo by propagation structures. In: 2015 IEEE 31st international conference on data engineering, IEEE, pp 651–662
10. Castillo, C., Mendoza, M., & Poblete, B. (2011, March). Information credibility on twitter. In Proceedings of the 20th international conference on world wide web, pp 675–684
11. Yang F, Liu Y, Yu X, Yang M (2012) Automatic detection of rumor on sina weibo. In: Proceedings of the ACM SIGKDD workshop on mining data semantics, pp 1–7
12. Kwon S, Cha M, Jung K, Chen W, Wang Y (2013). Prominent features of rumor propagation in online social media. In: 2013 IEEE 13th international conference on data mining, IEEE, pp 1103–1108
13. Yang Y, Niu K, He Z (2015) Exploiting the topology property of social network for rumor detection. In: 2015 12th international joint conference on computer science and software engineering (JCSSE), IEEE, pp 41–46
14. Chaudhary A, Mittal H, Arora A (2019) Anomaly detection using graph neural networks. In: 2019 international conference on machine learning, big data, cloud and parallel computing (COMITCon), (pp 346–350), IEEE
15. Giasemidis G, Singleton C, Agrafiotis I, Nurse JR, Pilgrim A, Willis C, Greetham DV (2016) Determining the veracity of rumours on Twitter. In: International conference on social informatics, Springer, Cham, pp 185–205
16. Kwon S, Cha M, Jung K (2017) Rumor detection over varying time windows. PLoS ONE 12(1):e0168344
17. Wang WY (2017) liar, liar pants on fire: a new benchmark dataset for fake news detection. arXiv preprint arXiv:1705.00648
18. Khanam Z, Alwasel BN, Sirafi H, Rashid M (2021) Fake news detection using machine learning approaches. In IOP conference series: materials science and engineering 1099(1):012040, IOP Publishing
19. Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781
20. Kaliyar RK, Goswami A, Narang P (2021) FakeBERT: Fake news detection in social media with a BERT-based deep learning approach. Multimedia Tools Appl 80(8):11765–11788
21. Devlin J, Chang MW, Lee K, Toutanova K (2018). Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805
22. Ma J, Gao W, Mitra P, Kwon S, Jansen BJ, Wong KF, Cha M (2016) Detecting rumors from microblogs with recurrent neural networks
23. Ruchansky N, Seo S, Liu Y (2017) Csi: a hybrid deep model for fake news detection. In: Proceedings of the 2017 ACM on conference on information and knowledge management, pp 797–806
24. Khattar D, Goud JS, Gupta M, Varma V (2019) Mvae: multimodal variational autoencoder for fake news detection. In: the world wide web conference, pp 2915–2921
25. Agarwal A, Mittal M, Pathak A, Goyal LM (2020) Fake news detection using a blend of neural networks: an application of deep learning. SN Comput Sci 1(3):1–9

26. Tuan NMD, Minh PQN (2021). Multimodal fusion with BERT and attention mechanism for fake news detection. arXiv preprint arXiv:2104.11476
27. Jwa H, Oh D, Park K, Kang JM, Lim H (2019) Exbake: automatic fake news detection model based on bidirectional encoder representations from transformers (bert). Appl Sci 9(19):4062
28. Zhang T, Wang D, Chen H, Zeng Z, Guo W, Miao C, Cui L (2020) BDANN: BERT-based domain adaptation neural network for multi-modal fake news detection. In: 2020 international joint conference on neural networks (IJCNN), IEEE, pp 1–8
29. Srivastava Y, Khanna P, Kumar S (2022) COVID-19 spread: a demographic analysis. In: International conference on innovative computing and communications, Springer, Singapore, pp 497–507
30. Glazkova A, Glazkov M, Trifonov T (2021) g2tmn at constraint@ aaai2021: exploiting CT-BERT and ensembling learning for COVID-19 fake news detection. In: International workshop on combating on line hostile posts in regional languages during emergency situation. Springer, Cham, pp 116–127
31. Gundapu S, Mamidi R (2021) Transformer based automatic COVID-19 fake news detection system. arXiv preprint arXiv:2101.00180
32. Bang, Y., Ishii, E., Cahyawijaya, S., Ji, Z., & Fung, P. (2021). Model generalization on COVID-19 fake news detection. arXiv preprint arXiv:2101.03841.
33. Wani A, Joshi I, Khandve S, Wagh V, Joshi R (2021) Evaluating deep learning approaches for covid19 fake news detection. In: International workshop on combating on line hostile posts in regional languages during emergency situition, Springer, Cham, pp 153–163
34. Goyal A, Puri K, Jain R, Nagrath P (2022) Forecasting rate of spread of covid-19 using linear regression and LSTM. In: International conference on innovative computing and communications. Springer, Singapore, pp 123–134
35. https://data.mendeley.com/datasets/zwfdmp5syg/1

Review on Edge Computing-assisted d2d Networks

P. S. Niveditha, Saju P. John, and Serin V. Simpson

Abstract Device-to-device communication is an innovative paradigm which enables user equipment to communicate directly with other user equipment with or without the involvement of network infrastructure. It is an inevitable part of the Internet of Things. Hence, it makes wireless networks more spectrum and energy-efficient with traffic offloading. However, the massive growth of number of devices and the corresponding heavy data traffic generated at the edge of the network created additional burdens on the cloud computing due to the bandwidth and resources scarcity. Hence, edge computing is emerging as a novel strategy that brings data processing and storage near to the end users, leading to what is called edge computing-assisted device-to-device communication. This paper conducts a comprehensive survey on different techniques developed to enhance the performance of device-to-device networks by enabling edge computing capability for the devices in the communication network.

Keywords Device–to-device communication · Edge computing · 5G communication

1 Introduction

Device-to-device communication abbreviated as d2d communication is an advanced data transmission technology, which was developed in the motive of improving the communication network efficiency. In the LTE direct technology, the d2d-enabled

P. S. Niveditha (✉)
APJ Abdul Kalam Technological University, Thiruvananthapuram, Kerala, India
e-mail: nivedithapsukes@gmail.com

S. P. John
Department of Computer Science and Engineering, Jyothi Engineering College, Cheruthuruthy, Thrissur, Kerala, India

S. V. Simpson
Department of Computer Science and Engineering, SCMS School of Engineering and Technology, Ernakulam, Kerala, India

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_4

devices can communicate directly with each other, obeying a set of standard secure routing protocol. Device-to-device communication is a promising solution for the existing problems in networks such as spectrum utilization in the next generation of cellular network. The d2d communication provides a lot of advantages to the network such as high data transfer rate and reduced delay by maintaining the required QoS constraints.

Introduction of d2d communication was made after LTE release 12 in the 4th-generation communication network. The d2d maintains its power level based on the cellular uplink power control in order to avoid the issues that may arise due to interference with the cellular base station. The d2d communication can provide better reuse of the radio spectrum as it shrinks the radio transmission to the direct device-to-device communication scenario. It also helps to reduce the transmission power and makes the network much more energy efficient. The d2d communication has got many advantages including congestion control, reduced delay, QoS guarantees, enhanced cell coverage, etc., but still there exist some challenges that need to be cured in efficiently deploying this technology.

Edge computing is an upcoming technology that has got burning importance nowadays. Edge computing allows the users to access cloud services within the range of the network. The main aim of edge computing is to reduce the latency by minimizing the spectrum between computing device and the storage location. The cloud computing platforms are brought closer to the user's equipment with the advent of edge computing technique. The basic components of edge computing network are:

- Edge devices: includes all types of devices in the network.
- Edge cloud: deployed in the base station.
- Public cloud.

Edge computing has an important role in collaborative computing. Collaborative computing is being used in scenarios from simple sensor networks to the highly complex automatic robotic functionalities.

By combining the features of mobile edge computing technique with the device-to-device communication technique, the performance of the resulting network shows a tremendous improvement. This will help to overcome the limitations of both the technologies. Traffic offloading or computational offloading is the main advantage of the integrated network. The devices which are draining out of energy and resource get a chance to recover their efficiency by sharing the burden of computational tasks with the willing co-nodes present in the communication network. Compared to the cloud computing, the edge computing technology can provide fast response to the device-to-device communication-enabled devices which further reduces the execution delay.

Figure 1 shows the system architecture of d2d-enabled edge computing network. It combines the features of both device-to-device communication and the edge computing technology in order to attain certain quality requirements of the communication network. The d2d devices form a communication network and communicate directly within their communication range. They form a d2d link, represented by red arrow in the figure, for the direct communication.

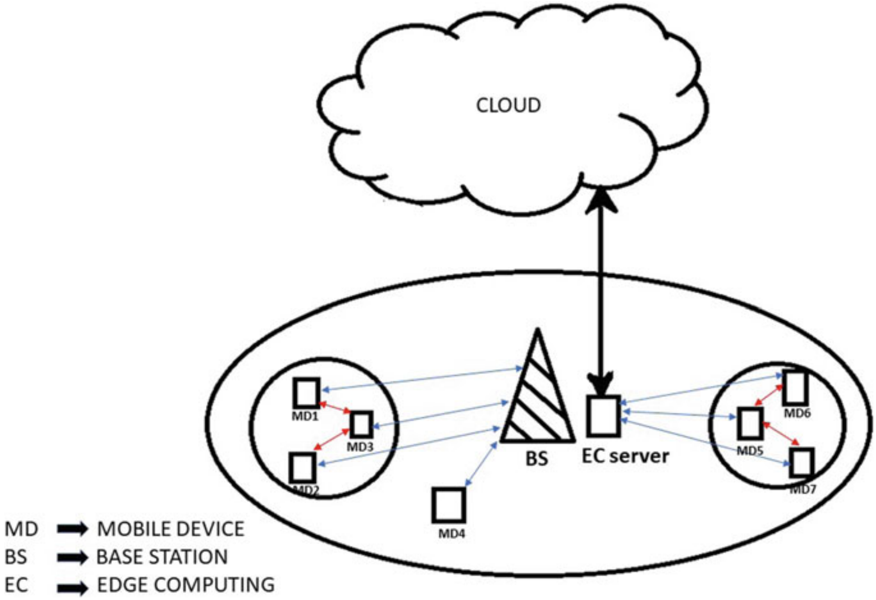


Fig. 1 System architecture of d2d-enabled edge computing network

The remaining sections of this review paper are organized as follows. The second section provides an analysis on the existing challenges in the device-to-device communication network and the challenges faced by the edge computing network. The third section provides a review on the various techniques to enhance the performance of edge computing-assisted device-to-device communication network, and finally, we summarize the conclusions in the fourth section.

2 Challenges in Existing Techniques

2.1 D2D Communication

Due to the low complex nature of the d2d devices, they have got widely accepted. The applications they are being used are also increasing day by day. But, it has got some challenges in aspects such as device discovery, mode selection, resource management, mobility-related issues, security, privacy, economic constraints some of the challenges are discussed here:

a. Device Discovery:
In d2d communication, the member nodes communicate directly. They do not have a central coordinator to coordinate the communications. The device nodes themselves

will have to discover the nearby node to communicate. In the distributed device discovery method, the devices themselves have the permission to discover other d2d users without including the base station. The time and energy required for the device discovery should be efficiently managed.

b. Interference Management:
Interference issues from the cellular users are a major challenge in d2d communication. Since, the d2d users and the cellular users will have to share the same resource in inband communication scenarios. The chances of getting interfered are very high in such situations. So, efficient interference methods have to be employed.

c. Security and Privacy:
Security and privacy threats are being identified in every communication network, especially in the wireless networks. Some of the important security threats that the d2d networks have to face are authenticity of users, confidentiality of data and user-related information, integrity, non-repudiation, secure routing and transmission, dependability, etc.

d. Power Control:
Power control has to be maintained at the edge user during the uplink communication and at the base station during the downlink communication. Various factors such as path loss, interference, number of units, delay, etc. highly effect the energy of the devices. It is very important to maintain power consumption at the optimal value in order to save energy resource.

e. Mode Selection:
In the d2d cellular network, the devices can communicate directly. This further improves the network throughput and reduces the delay in communication. There are mainly four modes of communication in the d2d communication, namely pure cellular mode, partial cellular mode, dedicated mode, and underlay mode.

2.2 Edge Computing

Even though edge computing has got a wide range of applications, it has got a few challenges to overcome for providing better performance. Some of the challenges faced by edge computing technology are:

a. Security:
Security is an issue that has got burning importance in the mobile edge platform. The chances of intrusion are very high in edge computing, and hence, it is very important to protect the applications and the data that are present in the edge server from being lost or leaked.

b. Mobility Management:
The devices in the wireless communication network are highly mobile. This results in the frequent path breaks. So, it is important to implement a mobility management

technique which helps to prevent the disconnection of the link between the edge network and the mobile devices.

c. Standard Protocol:
A standard protocol is essential to integrate different devices and applications working in variety of environments into a single compatible platform, where all the devices can work comfortably. Standard characteristics must be developed from the standard protocol.

d. Scalability:
The number of mobile devices present in the communication network is increasing every second. So, the edge computing technique being used should be scalable. That is, it should be able to provide services to all the needy users without any fault. The traffic should be maintained by the server for avoiding the bottleneck issues.

3 Edge-based Performance Enhancement in d2d Networks

The notion of methods to improve the performance of d2d devices has been a subject of interest for several years. Many researchers have discussed and put forward a wide variety of techniques as follows.

The works can be categorized on the basis of the results obtained as below:

3.1 Improvement of Computational Capacity

3.1.1 Proper Task Scheduling

Xie et al. [1] proposed a peer-to-peer supported task scheduling network to reduce the time requirement of edge computing-assisted d2d networks. They have proposed different modes of operations, namely mobile edge computing mode, hybrid mode, and P2P mode. In MEC mode, based upon sorting algorithm and some optimization technologies, a novel iterative algorithm is presented. This algorithm helps to find an optimal solution to the nonlinear programming optimization problem. A second mode is also defined, which is P2P mode. In P2P mode, the original problem is converted to a one-to-one assignment problem, which is further resolved using the Hungarian method of task scheduling. The third mode of operation is the hybrid mode, in which the solutions are found by incorporating linearization technology.

3.1.2 Integration of Access Point Nodes

Hu et al. [2] have proposed a performance analysis method for the edge computing-assisted device-to-device communication networks. This method makes use of two computing nodes, namely device-to-device node and the access point nodes. The end users are enabled to find a nearby access point or device-to-device node to offload their abundant computational tasks. They have also proposed a prioritized channel access policy to reduce the unwanted interferences among the mobile nodes present in the device-to-device communication network. Prioritized channel access policy divides the available channel for both uplink and downlink into two sets, in which one is reserved for the cellular users and the other set is shared between both the device-to-device links and the cellular links. The employment of this technology effectively improves the edge computing probability.

3.1.3 Edge-based Approach

Ateya et al. [3] have presented an efficient method to overcome the major issues in 5G cellular system such as traffic, security issues, and capacity issues by combining device-to-device communication and edge computing technology. By developing these two technologies, the system can improve its overall capacity as well as reduce its traffic-related issues. The proposed system divides the entire structure into four parts, namely cloudlet tier, microcloud tier, mini cloud tier, and the main cloud tire. Cloudlet tier consists of the gateway devices and the master device. Gateway device can be considered as a powerful user equipment with wide range of capabilities, whereas the master device processes the data provided the gateway devices in order to generate the desired result. The second part is the microcloud tier, which consists of the mini cloud units and the microclouds. The third and fourth sections are the mini cloud tier and the main cloud tier which are the higher levels of the previous two units.

3.1.4 Enhanced Traffic Offloading

Yang et al. [7] have proposed a novel technique in data traffic offloading scenario. In the other existing techniques, the device-to-device nodes can offload their traffic burden only with the nodes which are closer to them obeying various distance constraints. Whereas in the proposed system, they have implemented an incentive system which promotes the nearby nodes to exhibit their willingness to offload the traffic in desire of the incentive. These nodes also share the data and information about the offloading task to their adjacent mobile nodes too. This helps in increasing the reachability of the offloading requirements and reduces the delay in the offloading tremendously. It helps to find the nodes with lower computing cost in carrying out the required task.

3.2 Resource Allocation

3.2.1 Optimal Time Reuse Profile

Shengda et al. [4] proposed a low complex algorithm ADARP to reduce the computational burden of a single node by distributing it all over the network. It is semi-distributed time reuse scheme. They make use of the fog computing technique to solve the heavy traffic in the network, which is tremendously increasing every second. The d2d communication links are regarded as an important feature of proximity-based next-generation communication network. These d2d user equipment are used as fog nodes in this proposed paper. Various tasks are distributed among the fog nodes, which further reduce the computing complexity. They have proposed a distributed resource allocation scheme which maximizes the path-finding metric. They have implemented the algorithm for the optimal association of the fog nodes, and the study results show that this algorithm converges very quickly without sacrificing the performance quality.

3.2.2 Task Offloading Scheme

Feng et al. [5] proposed a novel technique to offload the traffic in the device-to-device network by tactfully sharing the computational tasks with the nearby and highly capable mobile devices. They have proposed a new algorithm named as SEEP. The main advantage of the SEEP algorithm is that it does not need prior knowledge regarding the mobile node and its current status. As the edge computing resides at the end of the network, it helps to minimize the execution delay and provides smooth computation offloading. The steerable economic expense algorithm is based on the Lyapunov algorithm. Resource-rich mobile devices distribute their available computational resource for those mobile devices which are in scarce of resource to carry out the computational tasks. They have also proposed an expense minimization procedure by optimizing the power allocation to the nodes, task scheduling, etc. The proposed algorithm establishes through its simulation result that it is capable of smoothly handling the tradeoff between delay and the expense.

3.2.3 Overall Computation Latency

Xing et al. [8] proposed a novel method for solving the latency-related issues in computational offloading. They have proposed an efficient algorithm for this. The proposed algorithm is implemented in an energy-efficient manner. This system makes use of the multihelper mobile edge computing system in which the d2d device, which needs the computational offloading shares the tasks among the multiple helpers who are willing to carry out the tasks. Then, the results are collected from the helper nodes at the predefined time intervals. The time intervals are selected based on the time division multiple access technique. This helps to reduce the interference-related issues as well as the chances of the data packets getting collided. The helpers send their computational results only after confirming that their computation is completed and that the channels are available for downloading the data.

3.2.4 Low Complex Algorithm is Proposed

Li et al. [10] proposed a novel technique to enhance resource utilization and the capacity of the network. Proposed system consists of a number of smart devices and wireless access points. The smart devices choose the apt wireless access point based on certain constraints and start transmitting data to the edge cloud through that selected access point. They have proposed a novel optimization algorithm which consists of two phases. The first phase deals with the relay selection, and the second phase deals with the resource allocation. Lagrange method is used for finding the optimum solution for resource allocation. It is an energy-efficient method which also maintains the network capacity. They have defined a step-by-step procedure in defining the algorithm to generate the feasible solution. Relay plays an important hand in the traffic offloading procedure without sacrificing the energy constraints.

3.2.5 Secured Network Access

Wang et al. [16] have proposed a novel technique to dynamically handle the communication resource and to detect the network structure without human interventions. It is made possible by incorporating the knowledge obtained from d2d devices for handling large number of device-to-device communication users. In order to support flexible network access, the following features such as energy, storage, provided services, resources are virtualized. It is made possible by integrating multidisciplinary theories including management, sociology, psychology, etc. The virtual management layer helps to create the optimal data transmission path. Resource management is a cumbersome process in communication networks. The proposed knowledge-centric edge computing technique is a dynamic method to manage and generate the resources efficiently.

3.3 Computational Offloading

3.3.1 Reduction in Consumed Energy

Diao et al. [12] have proposed a novel method to overcome the connectivity issues in the upcoming fifth-generation networks by incorporating non-orthogonal multiple access techniques, device-to-device communication technique, and edge computing technique. They have proposed a method to elevate the amount of computation resources available at the edge nodes. In order to optimize the resource allocation, they have proposed an adaptive algorithm. Particle swarm optimization technique is utilized to generate an effective power allocation algorithm. Based on swapping operations, they have derived an optimal channel allocation technique. At the end of the paper, they have proven that the proposed system was capable of reducing the delay to the minimum with the lowest energy consumption, through their simulation results.

3.3.2 Deals with the Ultra-dense Networks

Seng et al. [15] described a technique to solve the issue of computational offloading in the ultra-dense wireless communication networks. The user equipment will have to offload large amount of data to the edge servers and vice-versa. It is difficult to coordinate the data transfer between the user equipment and the edge network as the number of similar communications going on is extremely high. To solve this problem, the authors have suggested a decentralized method of offloading based on the blockchain technique. The proposed technique consists of two steps. In the first step, a request is generated by the user equipment to find the appropriate edge server. It is done based on the blockchain platform. The second step is to develop a user matching algorithm to find the compatibility of the computational task to be carried out with the edge server. The main constraints considered while making the decision are energy-related constraints and the execution time-related constraints.

3.3.3 Computation Offloading Over Multiple Edge Servers

Guo et al. [17] have proposed a computation offloading method in highly dense IoT networks. The number of devices in the IoT network is increasing every second, and along with that, the computational tasks are also increasing. In this scenario, it is very important to efficiently manage the resources without causing any conflict. The method employed to reduce the heavy traffic and the computational burden is offloading the task to edge servers, which are present in the radio access infrastructures such as in small cells and macrobase stations. They have proposed a greed offloading method as a solution for the highly dense network.

3.3.4 Effective in Heavy Traffic Conditions

Hou et al. [18] have proposed a novel method to reduce the computational burden in the devices. It incorporates the mobile edge computing technique in the d2d communication technique. Existing communication schemes ignore the greedy and selfish attitudes shown by the d2d users. The proposed scheme describes a hybrid offloading method. It includes two offloading modes: the cellular mode of offloading and d2d mode of offloading. Proposed scheme considers the following selection of the working mode, offloading ratio, resource management, d2d pair selection, etc. The authors have introduced the JCRO algorithm to optimize the resource allocation and the heavy task offloading. Optimization procedures help the request users to easily find their partners to offload the heavy computations.

3.3.5 Reduced Delay

Zhang et al. [19] have proposed a scheme in which the mobile devices share their resources with the devices that are in scarce of the resource. The authors have combined the functionalities of d2d communication in edge computing technology. Generally, the mobile devices are reluctant to accept the offloaded tasks from other users. But by including the auction scheme, the authors have helped to reduce the refusing nature of the devices and make the offer more attractive. The auction scheme mainly consists of three steps. The first step is to identify the willing nodes to offload their tasks and to confirm their identity. The second step is candidate selection. The third step is matching and pricing, in which both the willing node and the requested node match the computational tasks and agree on the pricing amounts fixed for the task.

3.3.6 Reduces the Inter-cell and Intra-cell Interferences

Wen et al. [20] have proposed an energy-efficient d2d system for the edge computing networks. The number of wireless communication devices is increasing tremendously, and as a result, the data hunger is also increasing. In such a scenario, it is very important to develop a scheme that provides cloud-based services. Edge computing is a promising technique which brings the cloud services close to the end users. But, it has got a lot of limitations such as limited resources, power supply constraints, low computing capacity. The authors have proposed a scheme which helps to overcome these limitations of edge computing and enhance its performance by incorporating a d2d offloading mechanism in edge computing. They have also proposed a novel technique to reduce the intra-cell and inter-cell interferences.

3.4 Power Constraints

3.4.1 Improves the Data Computational Rates

Wu et al. [11] described a wireless system to charge the users with the help of energy transmitting multi-antennas. It makes use of energy beamforming to charge the user nodes. They have also incorporated the offloading technique in which the user nodes can offload their data to nearby users also called as helper nodes. The system makes use of the frequency division multiple access technologies for the communication of the various d2d pairs. Different frequency bands will be provided for each user pairs so that it reduces the effect of inter-channel interference. It consists of the number of time frames. Each frame consists of three sets of time slots. One slot is meant for task execution, the second one for remote computing, and the final slot for result downloading from the helper nodes. Two optimization techniques, namely convex optimization and alternating optimization, are proposed to solve the issues related to the low computational rate.

3.4.2 Maximizes the Effective Capacity

Zhang et al. [6] have proposed a collaborative device-to-device communication scheme over the edge computing networks, maintaining the delay bounded QoS constraints. Existing techniques have laid back a few challenges behind, in the areas like multimedia data transmission, sustainable maintenance of QoS standards, maximizing the offloaded cellular traffic, reduction of interference to the nearby device-to-device communication users, etc. All these issues are solved to a certain extent in this paper. They have proposed device-to-device communication-matched algorithm to solve the issues related to capacity optimization. The algorithm is based on the bipartite graph. Both centralized and decentralized methods have been developed for the algorithm. Through the simulation results, they have proven that the proposed method is much better than the existing schemes in providing better QoS constraints.

3.5 Device Discovery

3.5.1 Incorporates MEC in D2D for the Efficient Device Discovery

Nikolova et al. [9] proposed a method which makes use of edge computing technique in device discovery and setup the device-to-device communication network. The mobile edge computing technique provides better working environment for the devices to work. With the advent of the edge computing, the computational tasks can be brought closer to the d2d devices, thereby reducing the processing delay and helps to reduce the energy consumption tremendously. Generally, the traffic offloading is initiated by the device, whereas in the proposed scenario, the offloading is initiated by a particular application. The traffic offloading control is given to some authorized applications. That application further finds the proximate devices for the traffic offloading. The application considers various aspects such as location of device, QoS, congestion in the network before carrying out the offloading procedure. It is a proximity-based service.

3.6 *Latency Reduction*

3.6.1 Reduction in Backhaul Traffic

Wu et al. [13] have developed a new technique to reduce the rush of heavy traffic present in the backhaul network. Backhaul network comprises the links between the core network and small subnetworks at the edge of the networks. Device-to-device communication can effectively help to reduce the traffic in the backhaul network by sharing it with the nearby user nodes. It is very essential to have a good traffic in order to receive a good-quality video. Self-parameterization technique is utilized to develop a low complex algorithm for optimizing the quality of experience of the streaming videos. The result helps the future designers to enable d2d communication into edge computing technique in the upcoming 5G communication. The adaptive video streaming technology provides low latency and high SNR to the mobile user nodes.

3.6.2 Achieves Low Delivery Latency

Karasik et al. [14] described a novel technique to make the edge user nodes more efficient by enabling cooperative transmission and reduce the interference management. A pipelined transmission is explained in which simultaneous transmissions are permitted either from edge network or from device-to-device communication network. The edge networks can receive data from d2d devices with a minimum delay as they are located at the edge of the communication network. A compress and forward device-to-device communication technique is presented to optimize the delivery time. The implementation of device-to-device communication shows a drastic improvement in the delivery latency as compared to the networks which have not employed d2d communication. By decreasing the traffic in the fronthaul link, it is possible to reduce the traffic load in the communication network.

Table 1 given shows the papers based on the problems considered in each.

4 Conclusions

This paper discusses various techniques employed in order to enhance the performance of edge computing-assisted d2d network. The challenges faced by both device-to-device communication and the edge computing techniques are discussed. By evaluating the various techniques employed in integrating device-to-device communication and edge computing shows that the performance has improved tremendously in aspects such as traffic offloading, reduction of delay, energy optimization etc.

Table 1 Comparison of various d2d-assisted edge computing techniques

Sl. No.	References	Objective	Technology employed	Advantages	Problems considered
1.	[1]	To minimize average task duration	P2P enhanced task scheduling	• Works in different modes in various situations • Simplifies original optimization problem • Optimal solution with lower complexity	Computational capacity
2.	[2]	To improve edge computing probability for users	Prioritized channel access policy	• Improves total successful edge computing • Useful in employing d2d-assisted edge computing networks	
3.	[3]	To overcome the capacity and traffic-related challenges in 5G network	Employs d2d with multilevel edge cloud units	• Offloads the core communication network • Improves the overall capacity of the network • Highly efficient in computational offloading	
4	[7]	To reduce computation offloading in d2d networks	Incentive propagation mechanism	• Computational offloading is not limited to nearby devices • Provides the minimum latency data propagation mechanism	

(continued)

Table 1 (continued)

Sl. No.	References	Objective	Technology employed	Advantages	Problems considered
5.	[4]	To overcome the time reuse problem	Alternate distributed allocation of time reuse patterns	• Identifies the optimal time reuse profile • Useful in fog-enabled in 5G networks • Uses minimum number of reuse patterns	Resource allocation
6.	[5]	To develop an effective computation execution mechanism	Steerable economy expense algorithm	• Efficient task offloading scheme • Method with minimum expense compared with various existing techniques	
7.	[8]	To conserve energy along with reduction of computation latency	The d2d-enabled multi-helper MEC system	• Minimization of overall computation latency considering individual energy • Handles the computational capacity constraints	
8.	[10]	To improve the capacity of the network and resource allocation	The d2d-enabled MEC system using JOSR algorithm	• Makes use of wireless AP • Ensures energy saving • Low complex algorithm is proposed	
9.	[16]	To develop a system which efficiently manage the communication resources dynamically	Knowledge-centric edge computing (KCE)	• Efficiently manages the communication resources • Secured network access with the proposed technique	

(continued)

Table 1 (continued)

Sl. No.	References	Objective	Technology employed	Advantages	Problems considered
10	[12]	To achieve the joint optimization	Scheduling-based joint computing resource power and channel allocation algorithm (S-JCRPCA)	• Reduces the delay of all the end users • Reduction in consumed energy • Provides optimization for power, resource, and channel allocation	Computational offloading
11	[15]	To develop a system to coordinate the computation offloading in ultra-dense networks	Decentralized computation offloading platform based on blockchain	• Deals with the ultra-dense networks • Proposed algorithm simplifies the selection of offloading partner with desired characteristics	
12	[17]	To improve the computation offloading in ultra-dense IoT networks	Mobile edge computation offloading (MECO)	• Address the resource hunger in IoT-based systems • Points the ultra-dense networks • Computation offloading over multiple edge servers	
13	[18]	A scheme to overcome the selfish and profit-seeking nature of d2d communication	Hybrid resource allocation and computation offloading scheme	• Overcomes the selfish and greedy nature of d2d networks • Highly effective in heavy traffic conditions	

(continued)

Table 1 (continued)

Sl. No.	References	Objective	Technology employed	Advantages	Problems considered
14	[19]	To overcome the refusing nature of d2d users against offloading tasks	Auction scheme for computation offloading scheme(ASCRA)	• Finds the effective way to offload heavy tasks • Reduced delay • It is budget-friendly	
15	[20]	To improve the energy efficiency in wireless communication	D2D incorporated mobile edge computing	• Reduces the inter-cell and intra-cell interferences • Better service is being provided to the end users	
16	[11]	To develop a wireless edge computing system with d2d task offloading	The d2d offloading of mobile edge computing system	• Improves the data computational rates • Helpful in the development of low-power wireless devices • Maximum utilization of energy and resources	Power constraints
17	[6]	To overcome the power allocation problems in heterogeneous delay bounded networks	Collaborative d2d caching model	• Maximizes the effective capacity • Have developed both the centralized and decentralized d2d caching algorithms	
18	[9]	To initiate an effective d2d communication by discovering the proximate devices	D2DCS with proximity-based service functions.	• Incorporates MEC in D2D for the efficient device discovery • It is described with the help of typical use cases	Device discovery
19	[13]	To ensure the quality of live streaming videos	Dynamic adaptive video streaming (DASH)	• Reduction in backhaul traffic • It maintains the average video bitrate per user	Latency reduction

(continued)

Table 1 (continued)

Sl. No.	References	Objective	Technology employed	Advantages	Problems considered
20	[14]	To reduce data delivery delay	The d2d-assisted fog networks with edge caching	• Details the uses of the out of band d2d network • Achieves low delivery latency • Reduces the traffic in fronthaul link	

References

1. Xie Z, Song X, Xu S (2020) Peer-to-peer enhanced task scheduling for D2D enabled MEC network. IEEE Commun Mag, vol 8
2. Hu H, Zong P, Wang H, Zhu H (2020) Performance analysis for D2D-enabled cellular networks with mobile edge computing variations, challenges, capacity and protocol issues IEEE Communi Mag
3. Ateya AA, Muthanna A, Koucheryavy A (2018) Multi-level edge computing framework for 5G cellular system with D2D enabled communication. In: 2018 Proceedings international conference on advanced communications technology (ICACT), pp 507–512
4. Jin S, Zhu Z, Yang Y, Zhou MT, Luo X (2017) Alternate distributed allocation of time reuse patterns in fog-enabled cooperative D2D networks. In: Proceedings IEEE fog world congress (FWC)
5. Feng J, Zhao L, Du J, Chu X, Yu FR (2018) Computation offloading and resource allocation in D2D-enabled mobile edge computing. In: 2018 Proceedings IEEE international conference
6. Hu YC, Perrig A, Johnson DB (2019) Heterogeneous statistical QoS-driven power allocation for collaborative D2D caching over edge-computing networks. In: 2019 Proceedings IEEE 39th international conference on distributed computing systems (ICDCS), pp 944-953
7. Yang L, Zhu H, Wang H, Qian H, Yang Y Incentive propagation mechanism of computation offloading in fog-enabled D2D networks
8. Xing H, Liu L, Xu J, Nallanathan A (2018) Joint task assignment and resource allocation for D2D-enabled mobile-edge computing. IEEE Trans Commun
9. Pencheva EN, Atanasov II (2018) Mobile edge service for D2D communications. In: 2018 proceedings XXVII international scientific conference electronics—ET2018
10. Li Y, Xu G, Yang K, Ge J, Liu P, Jin Z (2015) Energy efficient relay selection and resource allocation in D2D-enabled mobile edge computing. IEEE Trans Veh Technol 14(8)
11. Wu D, Wang F, Cao X, Xu J (2019) Joint communication and computation optimization for wireless powered mobile edge computing with D2D offloading. J Commun Inf Networks 4(4)
12. Diao X, Zheng J, Wu Y, Cai Y (2018) Joint computing resource, power and channel allocations for D2D-assisted and NOMA-based mobile edge computing. In: 2018 Proceedings IEEE Translations
13. Mehrabi A, Siekkinen M, Illahi G, Ylä-Jääski A (2019) D2D-enabled collaborative edge caching and processing with adaptive mobile video streaming. IEEE Trans
14. Karasik R, Simeone O, Shlomo S (2019) How much can D2D communication reduce content delivery latency in fog networks with edge caching. IEEE Trans Commun
15. Seng S, Li X, Luo C, Ji H, Zhang H (2019) A D2D-assisted MEC computation offloading in the blockchain-based framework for UDNs. In: Proceedings IEEE transactions
16. Wang R, Yan J, Wu D, Wang H, Yang Q (2018) Knowledge-centric edge computing based on virtualized D2D communication systems. IEEE Commun Magazine

17. Guo H, Liu J, Zhang J, Sun W, Kato N (2018) Mobile-edge computation offloading for ultra-dense IoT networks. IEEE Internet Things J
18. Hou J, Wang X, Wang D, Lan Y, Liu Z (2019) Computation offloading strategy in D2D-assisted cellular networks with mobile edge computing. IEEE/CIC International conference on communications workshops in China
19. Zhang R, Shi W, Zhang J, Liu W (2019) An auction scheme for computing resource allocation in D2D-assisted mobile edge computing. In: 2019 Proceedings IEEE international conference
20. Wen J, Ren C, Sangaiah AK (2018) Energy-efficient device-to-device edge computing network: an approach offloading both traffic and computation. IEEE communications magazine
21. Simpson SV, Nagarajan G (2021) SEAL—security-aware list-based routing protocol for mobile ad hoc network In: Priyadarshi N, Padmanaban S, Ghadai RK, Panda AR, Patel R (eds) Advances in power systems and energy management. ETAEERE 2020. Lecture notes in electrical engineering, vol 690. Springer, Singapore
22. Simpson SV, Nagarajan G (2021) A table based attack detection (TBAD) scheme for Internet of Things: an approach for smart city environment. In: 2021 International conference on emerging smart computing and informatics (ESCI), Pune, India, pp 696–701
23. Simpson SV, Nagarajan G (2021) A fuzzy based co-operative blackmailing attack detection scheme for edge computing nodes in MANET-IOT environment. Future Gener Comput Syst 125:544–563
24. Simpson SV, Nagarajan G (2021) An edge based trustworthy environment establishment for internet of things: an approach for smart cities. Wireless Networks, pp 1–17
25. Simpson SV, Nagarajan G (2020) SEAL—security-aware list-based routing protocol for mobile ad hoc network. In: International conference on emerging trends and advances in electrical engineering and renewable energy, Springer, Singapore, pp 519–530
26. Simpson SV, Nagarajan G (2021) A table based attack detection (TBAD) scheme for Internet of Things: an approach for smart city environment. In: 2021 International conference on emerging smart computing and informatics (ESCI), IEEE, pp 696–701

OpenDaylight SDN and NFV Integration in OpenStack Cloud: OpenSource Approach for Improving Network Services

Hicham Boudlal, Mohammed Serrhini, and Ahmed Tahiri

Abstract Introducing software-defined networking and network function virtualization has brought new opportunities in cloud, enabling dynamic and autonomous configuration as well as provisioning of resources in the cloud. In this article, we highlight how OpenDaylight can be integrated with OpenStack to enhance network services. After describing SDN with NFV and how they compromise cloud services, we present some advantages of SDN in cloud networks. In addition, the deployment of an SDN solution on a high-performing cluster is presented. Finally, this cluster is integrated into the cloud OpenStack to enrich its services.

Keywords Cloud computing · SDN · Network architecture · NFV · OpenDaylight · OpenStack · OpenFlow

1 Introduction

Cloud computing has taken the IT industry and service providers into a whole different era, redefining the way IT resources and services are provisioned and used. In cloud computing, separate and distributed hardware resources, such as storage and computing capacity, can be acquired and utilized on demand, providing low-cost scalability and elasticity for applications. Traditional cloud networks typically had combined control and data plane where router device decides what path to take the traffic through policy and routing algorithm.

Software-defined networking (SDN) and network functions virtualization (NFV) hold the key to making a significant impact in this area. SDN relies on separating network control of data forwarding functionality, which allows controllers to directly program and abstract the underlying infrastructure as an abstraction of the network's top-level service functionality and network applications. NFV provides an alternative approach to designing, deploying, and managing network resources. Separating

H. Boudlal (✉) · M. Serrhini · A. Tahiri
Faculty of Sciences, Mohammed First University, BV Mohamed VI, BP 717, 60000 Oujda, Morocco
e-mail: hicham.boudlal5@gmail.com

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_5

the functions of the network, like firewalls, threat identification, etc., from owned hardware devices to allow them to be used more efficiently. Proprietary hardware devices are used in software and are deployed where needed [1–4].

The structure of the paper is as follows: An overview of cloud computing is presented in Sect. 2. A network architecture enabling SDN is presented in Sect. 3. Following, benefits of SDN in the cloud are described in Sect. 4. We illustrate our proposed architecture in Sect. 5, the implementation of SDN and NFV in the OpenStack cloud and discussion. Conclusion and future work are outlined in Sect. 6.

2 Overview of the Cloud

For several years, a concept called cloud computing has been researched and practiced and has been classified into three categories of services: Platform as a Service (PaaS), Software as a Service (SaaS), and Infrastructure as a Service (IaaS). PaaS is based on the sub-layer, where developers of applications are able to use services of the platform delivered through the cloud. While SaaS delivers full-featured software to cloud clients, integrating cloud-based messaging services, networking services, scheduling services, and essentially all of the programs that run on cloud. IaaS is its most basic, providing virtual machine servers as well as the associated infrastructure to clients in the cloud. The infrastructure is used for any purpose by the buyers, for example to deploy their owned servers or to develop applications that run on the VMs. Both SaaS and PaaS are deployed when using IaaS. However, for cloud services to be delivered, this requires the provider to build and support a very large-scale cloud data center(s), where thousands or physical hosting devices are interconnected over thousands of switches in the network. Given the considerable complication of large-scale network connectivity, this requires a provider to look at the network in the data center through a different angle compared to the standard network [5–7].

3 SDN Enabled the Architecture of the Network

Software-defined networking (SDN) is an evolutionary concept that has changed the architecture of traditional networks. They actually separate the data plane of the control plane and place that in a central server called a controller. This removes network management complexity by placing it in the hands of a software controller, while providing abstracted control of the subjacent infrastructure. Both the data plane and the simple hardware components of the network are enabled, making it immediately programmable as well as being managed in a centralized environment [8]. Figure 1 shows the differences in concept of a traditional network and an SDN network. In the traditional model, a control plane is positioned above the routing plane within every device in a network, with discrete decision made by every commutator processing collected data from neighboring devices on a highly decentralized scale.

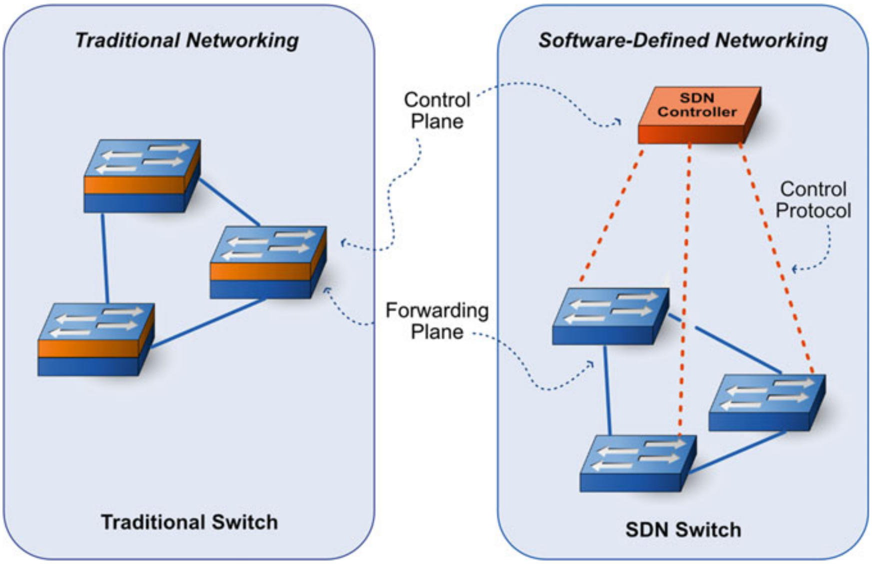


Fig. 1 Traditional networks compared to SDN

However, SDN puts the control plane in a centralized software controller that can monitor the network globally. Using information collected centrally, the controller is able to get simpler choices and deploy them to the transfer devices dynamically and efficiently [5, 9].

Open Networking Foundation (ONF) [10] proposed an SDN reference model as shown in Fig. 2 which is composed of three layers, that is, infrastructure layer, control layer, and application layer, stacked on top of each other.

The application layer includes programs communicating with a controller and requesting necessary resources. The control layer supplies logic-centric control functionality that monitors network behavior via opened interfaces. Typically, this is accomplished through a software component which can translate application requirements to data plane rules and provide the network information to them. The infrastructure layer is composed of the various components and low-level network devices involved in switching and transferring packets.

For interacting among different layers, there are two interfaces employed: the northbound interface, which assures that the applications communicate with a controller. The southbound interface ensures that the data path network elements communicate directly with the control layer, enabling the transfer rules to be controlled programmatically. The most extensive protocol used in this interface is OpenFlow [11].

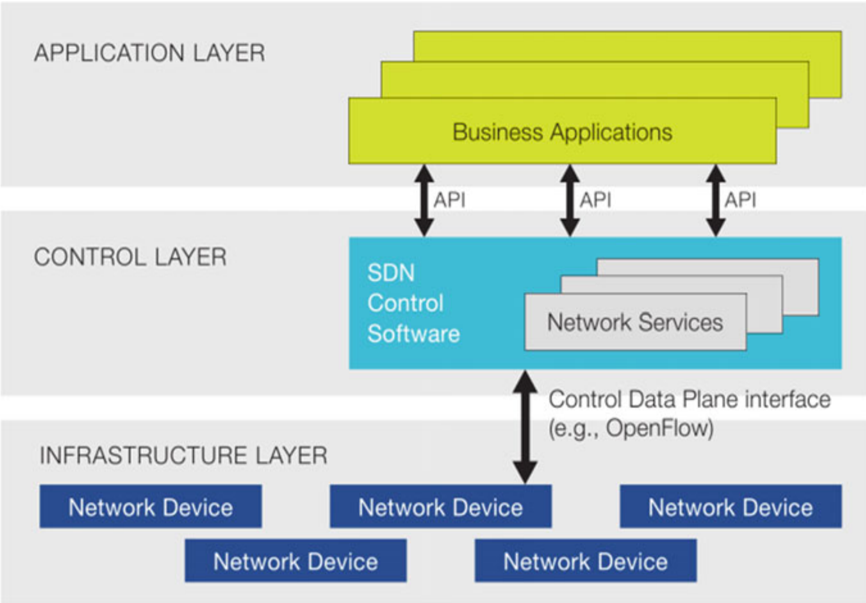


Fig. 2 SDN architecture

4 SDN Advantages in Cloud

Software-defined networking is the latest revolution in networking innovations [12], and cloud computing changes the way people have been doing computing and business [13]. Cloud networks are highly exposed and robust in providing services; therefore, while we could utilize SDN on a traditional network, there are multiple benefits to doing so.

Cloud-based centralized network provisioning: Not only can SDN along with programmability as well as virtualized technologies offer design flexibility, provisioning, and predefined scheduling, it provides the capacity for dynamic scalability to meet runtime requirements.

Integrated approach to business administration: With SDN, experimenting is possible inside virtualized network topologies and introducing novel alternatives is also streamlined and economical.

Intelligent security: With SDN, an easy and agile central application interface is provided for managing QoS policies, security, and controlling attacks.

Cloud abstraction: With SDN, network abstractions are essentially created to enable more agility and app-responsive behavior among a variety of cloud-IoT appliances. The network components comprising massive data center platforms are all manageable from the SDN controller.

Reduced downtime: SDN also supports the creation of configuration snapshots, allowing for quick recovery from upgrade failures.

Guaranteed content delivery: The capability to control and shape data traffic is a key benefit of software-defined networking. High-quality video streaming becomes easier as SDN provides improved network responsiveness to deliver a seamless user experience.

5 Proposed Architecture, Implementation, and Discussion

5.1 Proposed Architecture

An illustration of our proposed architecture is shown in Fig. 3 as an approach for improving network services, and the machine descriptions are as Table1.

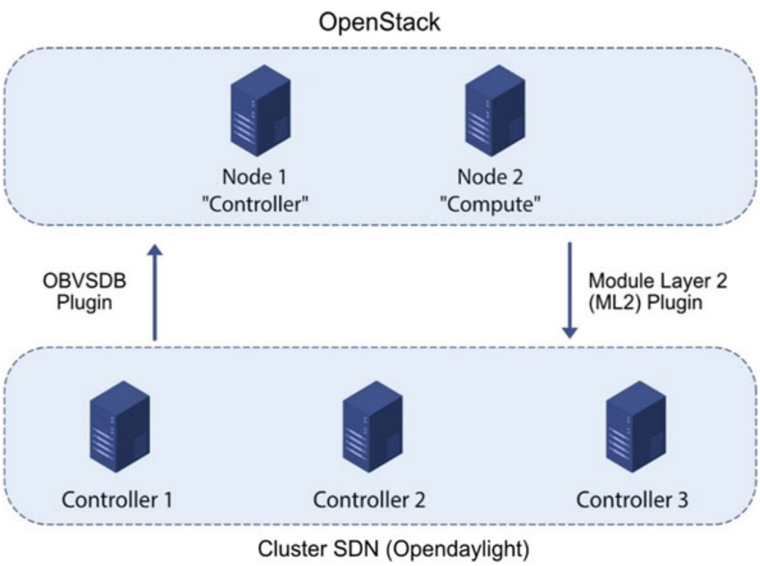


Fig. 3 Integration OpenStack and OpenDaylight

Table 1 Machine descriptions

Number of machine	Operating system	H/W detail	Purpose
1	Centos7	500 GB HDD 4 GB DR3 memory	Node 1 “Control”
1	Centos7	500 GB HDD 4 GB DR3 memory	Node 2 “Compute”
1	Ubuntu 14.04 X32	500 GB HDD 4 GB DR3 memory	Controller 1
1	Ubuntu 14.04 X32	500 GB HDD 4 GB DR3 memory	Controller 2
1	Ubuntu 14.04 X32	500 GB HDD 4 GB DR3 memory	Controller 3

5.2 Deploying an SDN Solution on a High-Performing Cluster

To ensure that the controller is more reliably available and to achieve the goal of high availability, we have opted to use OpenDaylight’s clustering technology. The installation procedure and the easy step-by-step installation are available at [14].

After running and configuring the three controllers in a cluster, we tested the cluster using a cluster monitoring tool. It will monitor the status of each node in the cluster, displaying two states—leader and follower. When a node is not working, it displays an error message. The leader state signifies that the node is in charge of this or that fragment, whereas follower is a safeguard state for a fragment that is under another node’s supervision.

Figure 4 shows the fragment for which the controller is responsible. The fragment names are displayed along X-direction, whereas controller roles appear along Y-direction. Follower and leader are displayed in the center of the window as shown in the figure. In this way, the status of the cluster can be monitored.

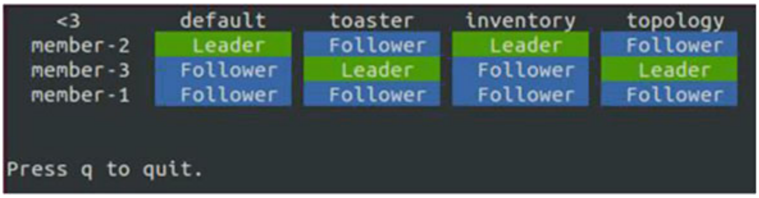


Fig. 4 Monitoring tool for clusters

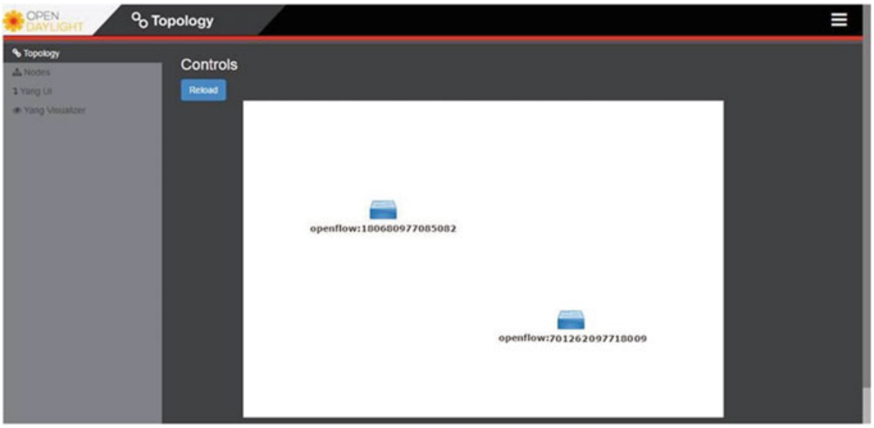


Fig. 5 DLUX showing Open vSwitch nodes

5.3 OpenDaylight SDN Cluster Integration with OpenStack Cloud Platforms

The OpenStack [15] can use OpenDaylight [16] as a network management provider via the Modular Layer 2 (ML2) plugin to the north. OpenDaylight handles network flows to OpenStack compute nodes via the OVSDB plugin in the south [17].

After the successful installation of OpenStack in two nodes (controller and compute), the installation procedure and easy step-by-step installation can be found in [18], and then, we moved on to its integration with the OpenDaylight cluster.

To achieve the integration, two main steps had to be taken: enabling OpenStack to utilize OpenDaylight as a provider of network management via the ML2 northward plugin and then OpenDaylight for managing the network flows for OpenStack nodes via the OVSDB southward plugin. After these steps, we got the following result. The two Open vSwitch of both compute and control nodes appeared on OpenDaylight as shown in Fig. 5.

Now, we have just to create and launch the instances. At this point, we confirm that the OpenStack integration is complete and the virtual machine traffic is controlled by OpenDaylight.

5.4 Discussion

The proposed architecture includes SDN and NFV plugin integrated into the OpenStack cloud network. SDN provides various new applications like resiliency, load balancing, and quality of service for cloud services to provide users for better flexibility and enhanced cloud service. Migrates traditional cloud network services into

SDN offers both flexibility and reliability through a programmable controller named SDN controller. NFV with SDN brings robust control of network through network virtualization as well as programmable routing of the network.

6 Conclusion and Future Work

The introduction of SDN has brought new varieties of approaches to the virtualization and monitoring of networks both within and external the cloud. In addition to NFV, SDN has enhanced cloud network control and expedited the delivery of novel network services. As an example of the applicability of integrating cloud computing and SDN paradigms, we demonstrated a practical study of OpenStack and OpenDaylight deployment. OpenDaylight and OpenStack are both large open-source projects that are backed by end-user and industry sponsored communities. Through our research, we have realized this insight. We incorporated SDN into cloud computing. Finally, it will be interesting to evaluate the overall performance of the proposal.

References

1. Manzalini A, Crespi N (2015) SDN and NFV for network cloud computing: a universal operating system for SD infrastructures. In: 2015 IEEE fourth symposium on network cloud computing and applications (NCCA), pp 1–6, IEEE, Munich. https://doi.org/10.1109/NCCA.2015.11
2. Battula LR (2014) Network security function virtualization (NSFV) towards cloud computing with NFV over Openflow infrastructure: challenges and novel approaches. In: 2014 international conference on advances in computing, communications and informatics (ICACCI), pp 1622–1628, IEEE, Delhi, India https://doi.org/10.1109/ICACCI.2014.6968453.
3. Singh S, Jha RK (2017) A survey on software defined networking: architecture for next generation network. J Netw Syst Manage 25:321–374. https://doi.org/10.1007/s10922-016-9393-9
4. Yi B, Wang X, Li K, Sk D, Huang M (2018) A comprehensive survey of network function virtualization. Comput Networks 133:212–262. https://doi.org/10.1016/j.comnet.2018.01.021
5. Son J, Buyya R (2018) A taxonomy of software-defined networking (SDN)-enabled cloud computing. ACM Comput Surv 51:1–36. https://doi.org/10.1145/3190617
6. Velte AT, Velte TJ, Elsenpeter RC (2010) Cloud computing a practical approach. McGraw-Hill, New York
7. Zhang Q, Cheng L, Boutaba R (2010) Cloud computing: state-of-the-art and research challenges. J Internet Serv Appl 1:7–18. https://doi.org/10.1007/s13174-010-0007-6
8. Rowshanrad S, Namvarasl S, Abdi V, Hajizadeh M, Keshtgary M (2014) A survey on SDN, the future of networking. JACST 3:232. https://doi.org/10.14419/jacst.v3i2.3754
9. Mishra V (2017) A survey on service-oriented network virtualization software defined networking for cloud computing. 12:7
10. Open Networking Foundation, https://opennetworking.org/.
11. McKeown N, Anderson T, Balakrishnan H, Parulkar G, Peterson L, Rexford J, Shenker S, Turner J (2008) OpenFlow: enabling innovation in campus networks. SIGCOMM Comput Commun Rev 38:69–74. https://doi.org/10.1145/1355734.1355746

12. Jain R, Paul S (2013) Network virtualization and software defined networking for cloud computing: a survey. IEEE Commun Mag 51:24–31. https://doi.org/10.1109/MCOM.2013.6658648
13. Xia W, Wen Y, Foh CH, Niyato D, Xie H (2015) A Survey on software-defined networking. IEEE Commun Surv Tutorials 17:27–51. https://doi.org/10.1109/COMST.2014.2330903
14. Setting Up Clustering—OpenDaylight Documentation Nitrogen documentation, https://docs.opendaylight.org/en/stable-nitrogen/getting-started-guide/common-features/clustering.html.
15. Open Source Cloud Computing Infrastructure, https://www.openstack.org/.
16. Platform Overview, https://www.opendaylight.org/about/platform-overview.
17. Open Daylight integration with OpenStack: a tutorial—Superuser, https://superuser.openstack.org/articles/open-daylight-integration-with-openstack-a-tutorial/.
18. Packstack—RDO, https://www.rdoproject.org/install/packstack/.

K-MNSOA: K-Anonymity Model for Privacy in the Presence of Multiple Numerical Sensitive Overlapped Attributes

Nidhi M. Chourey and Rashmi Soni

Abstract Knowledge is the main discussing and explored topic of today's era. Everyone is working toward improving information and tries to consider it as a ladder to move forward. Data is the main object to get information, and data is considered as a big data nowadays as it contains numerous information in all directions. As knowledge is bliss, it is also possible that an adversary can use this information to harm an individual. To protect data from an adversary privacy preserving data publishing techniques is used. But when multiple sensitive data present in a data set which is correlated to each other's several model are unable to protect data in an efficient way. In this paper, a novel model K-MNSOA is proposed for privacy preserving data publishing, which protect sensitive data privacy breach, even if the data set contains multiple sensitive numerical overlapped attributes. A proposed model assumes that all sensitive attributes are not actually sensitive, so when data is protected, information loss will increase. To overcome this issue, new model suggests to divide sensitive data into levels of sensitivity and apply generalization only for the privacy of high sensitive attribute.

Keywords Privacy preservation techniques · K-anonymity · Membership disclosure · Privacy breach

1 Introduction

As the dimensions of the data increase day by day, the need of privacy also increases [1]. Nowadays, data can be broadly divided into three categories: Structured data: Structured data is considered any data which is properly formatted or organized. Data sets or tables are the most common examples of structured data. When available data can be formatted in a similar manner and properly organized, we consider it as structured data, even if the sources of data are different [1].

N. M. Chourey (✉) · R. Soni
Department of Computer Science and Engineering, Oriental University, Indore, India
e-mail: nvchourey@gmail.com

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_6

Unstructured data: When data is collected from different sources and format of data varies from each other, such data is considered as unstructured data, or in other words, unorganized data. But to get specific knowledge from this type of data, we need to convert all data into one generic form. Semi-structured data: This data comes in between both abovementioned categories of data, i.e., data which is neither completely structured nor unstructured. It is a bit easier to convert this data in a specific relational form for analysis. In this work, we propose a model for structured data in the presence of multiple sensitive overlapped attributes present in the data set, where overlapped attributes mean keeping attributes together in a cell who are dependent in nature in anyways [2].

In the literature review section, all previously proposed models are discussed. In Sect. 3, new model K-MNSOA is proposed to overcome shortcomings of previously proposed models. Section 4 concludes this work and mentions future directions.

2 Literature Review

2.1 PPDPT: Need and Techniques

Privacy and security are most common and discussed topics from the last decades. One side we are able to expand our knowledge and capacities to information storage, and other side, privacy of this data concern increases [2]. Day by day we are believing to store data using cloud or at any other remote location, but this may increase concerns about privacy of data because information is easily disclosed in this aspect [1, 3, 4]. Information about any individual is available online, and on the basis of this, attacker can get extra information which is previously unknown [5]. The term PPDP is the used technique which is used to protect sensitive data against this type of disclosures. There are several techniques present for it, and K-anonymity is the most widely used technique [6]. Proposed model based on this technique also improves its shortcomings, i.e., membership disclosure and proximity breach details of both topics explained in the next sections. The need of information release occurs, when any organization needs to do surveys for any purpose, and if the information contains any sensitive data, then need of PPDPT arises, because when publicly available data is linked with this sensitive data, it may cause information disclosure of data [3]. In general term, data can be divided into three forms, and in data set terms, there are three types of attributes key, quasi-identifiers and sensitive attributes [7].

On the basis of these three attributes, disclosures are easily described as attribute, identity and membership disclosure [8]. To prevent these types of privacy breach, K-anonymity model was proposed by Sweeny [3]. The main objective behind this model was to control linking attack. Linking attack works when publicly available attributes are linked with any private organized data [2]. This may lead to disclosure of any previously unknown information.

2.2 K-Anonymity: Need and Techniques

When organizations are bound to release their confidential information, which contains all categories of data, K-anonymity techniques are used to protect sensitive data against linking attacks. There are many techniques associated with K-anonymity, but widely used technique is generalization and suppression [4]. In generalization technique, value of quasi-identifier is replaced with most general values or set of values, suppression technique is used for most susceptible data, and so, it is removed from the data set before release [7, 9, 10].

Attribute types are divided as follows: key attributes: name; quasi-identifier: gender, age, zip code, occupation, qualification and salary [11]; sensitive attributes: disease. When generalization is applied at age and zip code, K factor is satisfied. Generalization levels are only applied on selected attributes to achieve K factor to maintain the usability of data set [9]. The anonymity term means values which are unknown to others and causes confusion, because it converts data set values, very generalized and similar to other records. If K factor is 3, then for any record there is minimum two quasi-identifier sets which show same value sets [4].

2.3 L-Diversity

This model is proposed in 2006 to overcome drawbacks of K-anonymity. As a K-anonymity model worked so well against disclosure prevention and linking attacks, sometimes it fails to protect against background knowledge attack and diversity [12]. The background knowledge attack works based on prior knowledge about the individual. In this case, the attacker has some knowledge about person and wants to gain more knowledge which is unknown to him/her. If the anonymous data set not properly contains diversity, then attacker's motive is to infer sensitive data successes [13], for example if two records show same disease and the same attribute values: malaria. So if attacker has background knowledge about the individual, i.e., its occupation, then only he can be sure that the individual is suffering from malaria. But with the same case given, even if the attacker has no background knowledge about the individual, he can get information that only two options for a disease are present for the individual—malaria or flu. So his accuracy will be 50%. This shows lack of diversity. To increase diversity, generalization level is to be increased in extent that even if an attacker has strong background knowledge about the individual, he is unable to predict exactly disease [14].

2.4 Multiple Sensitive Attributes

If only one sensitive attribute disease is present in the data set, then all other attributes are considered as quasi-identifiers. So, to protect sensitive attributes, other attribute values of quasi-identifiers are generalized or suppressed. When data set contains multiple sensitive attributes, then prevention against disclosure becomes a tough task. Because, it increases the possibilities of membership disclosure and data breach [1, 7]. If a data set, which contains sensitive attributes as disease, occupation, qualification and salary and the table, shows diversity, even then the chances of membership disclosure increase, because if the adversary is successful to find one sensitive record, he is able to check another sensitive records associated with this record [8, 9].

2.5 Overlapped Attributes

Generalization technique is very effective to protect sensitive data against disclosures, but it also has some shortcomings [2]. Generalization technique reduces data utility because to achieve privacy in some cases, it suggests suppression technique, which leads to hiding data from data set. It also leads to correlation loss, i.e., it is also possible that some sensitive information corelated with another sensitive or non-sensitive attributes. For example, if Lupus disease is mostly present in dancers, but to prevent disclosure, it is suppressed. Because, occupation attribute is considered as a quasi-identifier. The scenario changes based on the attribute type, because suppression technique is applied only for quasi-identifiers. So to maintain correlation among two attributes, if we consider both as sensitive, then their correlation ship can be maintained. But in that case, the possibility of membership disclosure will be higher [5]. To solve these issues, two models were proposed in 2012: Slicing and 2016: k-AMOA. Both worked well to maintain a correlation among attributes, but were unable to protect against multiple numerical sensitive overlapped attributes [5]. K-AMOA model shows following advantages:

- It is for multiple overlapped attributes, to preserve correlations.
- Protection against membership disclosure, i.e., based on one disclosure, other sensitive information will not reveal.
- Bucketization technique is used where horizontal and vertical data portioning is applied.

Shortcomings of k-AMOA model:

- If multiple sensitive attributes show correlation among them, then the efficiency of model decreases [6].
- Not suitable for dynamic data set.
- Not suitable for multiple numeric sensitive attributes
- Attribute type classification was not clear

In proposed model K-MNSOA, to overcome this issue, we suggest to append one more quasi-identifier into correlated attribute set. So when generalization is applied on this quasi-identifier, possibility of member disclosure decreases and also it prevents against proximity breach, which arises in the case of multiple numerical sensitive attributes [1].

3 Proposed Model K-MNSOA: K-Anonymity Model for Multiple Numerical Sensitive Overlapped Attributes

3.1 Methodology

For this following points and questions to be answered:

Q.1 Model is based on which research design such as descriptive, analytical or experimental?
Ans. Experimental research design.
Q.2 What will be the study area and study population?
Ans. Medical data
Q.3 How to collect data and from where?
Ans. The experiment can be performed in previously proposed models data sets.
Q.4 What will the data analysis procedure?
Ans. Python or R can be used for implementation and analysis purpose.

On the basis of these decisions and objectives, the proposed algorithm is designed. Following are the objectives of proposed research work:

- To investigate a technique to reduce interaction of data publisher with anonymity technique, we assume that data publisher is a trustworthy one and has full knowledge over system, but sometimes data publisher is responsible for data breaches.
- To investigate a technique to allow data publisher, it has to decide attribute type, i.e., sensitive or non-sensitive.
- If the value of K is pre-defined, then the system will automatically modify techniques.
- If multiple numerical sensitive attributes are present, then the data set chances of proximity breach attack increase.
- If multiple corelated sensitive attributes are present, then chances of membership disclosure will be high.
- Slicing technique-based model was proposed for single overlapped attribute. So, single attributes overlapped in multiple columns may show challenges for security.
- Model should protect against membership disclosure effectively.

On the basis of these observations, problem statement can be stated as follows:

Given a data set D from private hospital PH that contains key attributes KA, quasi-identifiers QI and sensitive attributes SA. When released for some public use, data publisher removes KA and converts QI in anonymous form (A) and new data set becomes an anonymous data set AD. When multiple numerical sensitive attributes (M-SNA) are present in the data set, to achieve K-anonymity and diversity if generalization techniques are used, it affects correlations among attributes, and if the tuple suppression Technique is used, it causes data distortion. So, a new system needs to overcome this drawback also to control membership disclosure (MD) and privacy breach.

3.2 *Algorithm: K-MNSOA-K-Anonymity Model for Multiple Numerical Sensitive Overlapped Attributes*

Algorithm 1 shows step-wise procedure of proposed model. Tables 1 and 2 show input set and output of model.

Algorithm 1: K-MNSOAOA Algorithm
D—Data Set Collected From Organization For Applying PPDP
Tech nique
DI—Slice Of Data Set For PPDP
K—Min. Value For
Anonymization
MG—Maximum Generalization Value For QI
D—Min Diversity For Each Set, Generally Ld =
K-IAD-Attribute Division Technique
OA—Overlapped Attribute
SetQI—Quasi Identifiers
D2—Data Set Slice After AD
AG—Apply Generalization Technique On
QIAGL—Allowed Generalization Level
GL—Current Generalization Level
D—New Data Set After Ad Who Satisfies L—Diversity And
Proximity Breach, Divide// D—Into Si. Sn For Release//

Result: L-diverse K-anonymity Table
Save Data set D as D#;
Identify Attribute Type;
Remove KA;
Select QI's to apply Generalization;
while $(GS \leq KV)$

Table 1 Private table for K-MNSOA

S. No.	Name	Gender	Age	Zip Code	Disease	Weight	Occupation	Education	Salary	PF
1	Sam	M	22	50,022	Heart Issue	67	Manager	MBA	12	15,000
2	Meera	F	35	50,027	Hypertension	60	Teacher	BBA	5	20,000
3	Jack	M	20	52,346	Lupus	64	Dancer	BA	3	10,000
4	Mandy	M	39	52,340	Broken Arm	75	Doctor	MBBS	25	70,000
5	Suman	F	33	57,890	Flu	80	Nurse	Bsc	2	20,000
6	Tom	M	36	57,890	Malaria	66	Accountant	CA	15	50,000
7	Tommy	M	39	52,347	HIV	67	Manager	MBA	7	60,000
8	Suji	F	22	50,027	Cold	56	Student	B.Tech	4	12,000
9	Pooja	F	34	50,028	Fever	76	Professor	Ph.D.	10	40,000
10	Radha	F	20	52,340	Cancer	70	Manager	B.Tech	6	15,000
11	Ankita	F	36	57,899	Malaria	66	Accountant	CA	15	53,000
12	Dk	M	39	52,340	Flu	67	Manager	MBA	7	60,000

Table 2 Results of K-MNSOA

S. No.	Age, gender, disease	Zip Code, disease	Occupation, weight	Quali., salary	PF, ZipCode
8	$\leq 25, F, cold$	50,027, cold	Student, 56	B.Tech, 4	12,000, 50027
2	$\leq 35, F, hypertension$	50,027, hypertension	Teacher, 60	BBA, 5	20,000, 50027
3	$\leq 25, M, lupus$	52,346, lupus	Dancer, 64	BA, 3	10,000, 52346
1	$\leq 25, M, heartIssue$	50,022, heart issue	Manager, 67	MBA,12	15,000, 50022
6	$\geq 36, M, malaria$	57890, malaria	Accountant, 66	CA,15	50,000, 57890
7	$\geq 36, M, HIV$	52347, HIV	Manager, 67	MBA, 7	60,000, 52347
5	$\leq 35, F, flu$	57890, flu	Nurse, 80	B.sc., 2	20,000, 57890
9	$\leq 35, F, fever$	50028, fever	Professor, 76	Ph.D., 10	40,000, 50028
10	$\leq 25, F, cancer$	52340, cancer	Manager, 70	B.Tech, 6	15,000, 52340
11	$\geq 36, F, malaria$	57899, malaria	Accountant, 66	CA, 15	53,000, 57899
12	$\geq 36, M, flu$	52340, flu	Manager, 67	MBA, 7	60,000, 52340
4	$\geq 36, M, broken\ arm$	52340, broken arm	Doctor, 75	MBBS, 25	70,000, 52340

```
do
    Apply GL on QI;
    if (GL ≠ AGL)
     then
        Apply GL+ on QI;
        if (L≠ k1)
         then
            Apply GL+ on QI;
            if  (D# ≥ ProximityFactor)
             then
                goto next step;
            else
        else
            Apply GL+;
        end
        Check once again and save Data Set D2
    else
        Arrange data as per QI values;
        Divide D# Into S1......... Sn For Release;
    end
    Release S1......Sn;
end
```

Figure 1 shows flowchart of the algorithm. Proposed model privacy preserving technique for multiple corelated sensitive attributes will produce the following outcomes:

- Released data maintains correlations among attributes.
- System works effectively for parallel sequential published versions.
- Protection from proximity breach attack and membership disclosure in the presence of multiple numerical sensitive overlapped attributes.
- Comparative analysis of proposed techniques with the existing techniques.

4 Conclusion

The K-MNSOA model proposed a novel approach for multiple numerical sensitive overlapped attributes. As discussed in Sect. 2, prevention against membership disclosure and privacy breach is quite difficult for this type of data. Proposed model focused on protection against proximity breach very effectively. In the initial steps of model, it increases the dimensionality of data sets, and at the last step of the model, it reduces dimensionality. So the unnecessary data is not released. This work will be further extended in terms of working with dynamic data sets when multiple sets of overlapped attributes are present in data set.

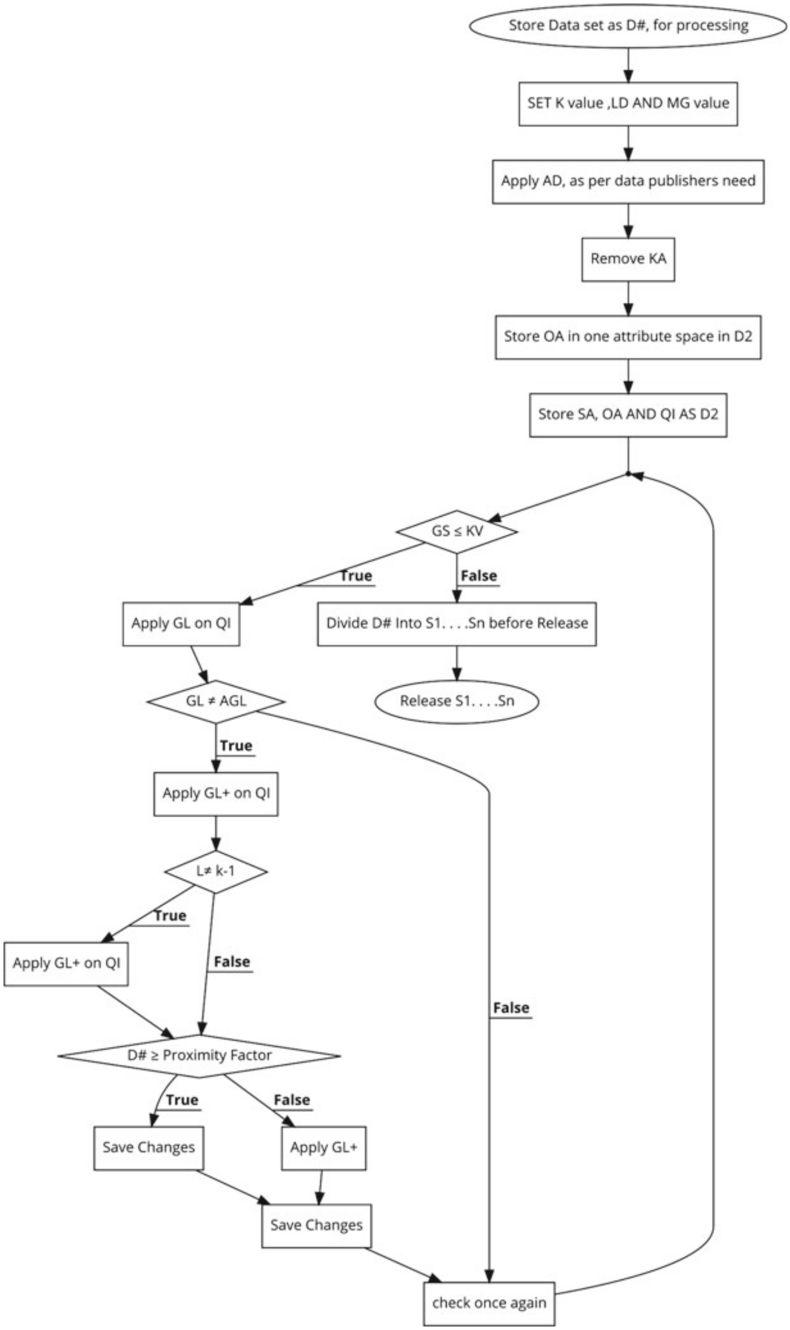


Fig. 1 Flowchart of K-MNSOA Algorithm

References

1. Liu Q, Shen H, Sang Y (2015) Privacy-preserving data publishing for multiple numerical sensitive attributes. Tsinghua Sci Technol 20(3):246–254. https://doi.org/10.1109/TST.2015.7128936, http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=arnumber=7128936isnumber=7128931
2. Maheshwarkar N, Maheshwarkar B, Patidar P, Rawat MK (2016) K-AMOA: K-anonymity model for multiple overlapped attributes, conference ICTCS '16, March 04–05, 2016, Udaipur, India© 2016 ACM. ISBN 978-1-4503-3962. https://doi.org/10.1145/2905055.290514
3. Sweeney L (2002) Achieving K-Anonymity privacy protection using generalization and suppression. Int J Uncertain Fuzziness Knowl Based Syst 10(5):571–588
4. Geetha R, Karthika S, Kumaraguru P (2021) Tweet-scan-post: a system for analysis of sensitive private data disclosure in online social media. Knowl Inf Syst 63:2365–2404. https://doi.org/10.1007/s10115-021-01592-2
5. Li T, Li N, Zhang J, Molloy I (2012) Slicing: a new approach for privacy preserving data publishing IEEE Trans KDE 24(3)
6. Sweeney L (2002) K-Anonymity: a model for protecting privacy. Int J Uncertain Fuzziness Knowl Based Syst 10(5):557–570
7. Samarati P, Sweeney L (1998) Protecting privacy when disclosing information: K-anonymity and its enforcement through generalization and suppression
8. Machanavajjhala A, Gehrke J, Kifer D, Venkitasubramaniam M (2006) l-diversity: privacy beyond k-anonymity. In: Proceedings of 22nd international conference on data engineering. (ICDE), p 24
9. Hu X, Sun Z, Wu Y, Hu W, Dong J (1989) K-anonymity based on sensitive tuples. In: 2009 first international workshop on database technology and applications, 978-0-7695-3604-0/09 /2009 IEEE. https://doi.org/10.1109/DBTA.2009; Young M (1989) The technical writer's handbook. University Science, Mill Valley, CA
10. Tesfay G, Serna J, Rannenberg K (2019) PrivacyBot: detecting privacy sensitive information in unstructured texts. In: 2019 sixth international conference on social networks analysis, management and security (SNAMS), pp 53-60. https://doi.org/10.1109/SNAMS.2019.8931855
11. Zhu N, Chen B, Wang S et al (2021) Ontology-Based Approach for the Measurement of Privacy Disclosure. Inf Syst Front. https://doi.org/10.1007/s10796-021-10180-2
12. G. Prabu Kanna and V. Vasudevan, "A fully homomorphic–elliptic curve cryptography based encryption algorithm for ensuring the privacy preserva- tion of the cloud data," Cluster Comput., vol. 22, pp. 9561–9569, 2019, doi: https://doi.org/10.1007/s10586-018-2723-9.
13. Y. Pu, J. Luo, Y. Wang, C. Hu, Y. Huo, and J. Zhang, "Privacy-Preserving Scheme for Location-Based Services Using Cryptographic Approach," Proc. 2018 2nd IEEE Symp. Privacy-Aware Comput. PAC 2018, pp. 125–126, 2018, doi: https://doi.org/10.1109/PAC.2018.00022.
14. K.SIvanna, S. Prabhu Deva, and M. Santoshkumar, "Privacy Preservation in Cloud Computing with Double Encryption," vol. 5, no. May 2017, pp. 149–156, 2017, doi: https://doi.org/10.1007/978-981-10-3226-4.

Modelling 5G Data Using Tree-Based Machine Learning Models

P. Mithillesh Kumar and M. Supriya

Abstract 5G or fifth generation is the latest in the communication technology which is being researched worldwide as a successor to the current 4G technology. 5G operates on higher bandwidth with higher data rates of the order of Gbit/s. 5G is estimated to play a major role in the development of smart cities and IoT use cases. Lumos 5G is one of the groups researching on the topic. In this paper, the throughput obtained under various conditions is analysed as a regression model in machine learning with the features as continuous variables. It is observed that the newer tree machine learning models are performing better on the dataset than the traditional tree models. This is verified by performing a tenfold cross-validation check on the best performing models.

Keywords 5G · Regression · Throughput · Trees · Cross-validation

1 Introduction

The need for higher bandwidths and higher data rates is increasing day to day with the rate of increase of Internet adoption and the increased need to be satisfied. Higher the bandwidth, higher is the rate of data transfer, analysis and the development of the society. The bandwidths are graded using data rates, and the developments are given by generations. The generations range from the first generation to the fourth generation. The current generation of data transfer being implemented in the cellular networks is called the fourth generation of wireless connection or Long-Term Evolution (LTE). This is turning insufficient with the rapid increase in the number of devices connected to the Internet and the network load generated these days.

P. Mithillesh Kumar (✉) · M. Supriya
Department of Computer Science and Engineering, Amrita School of Engineering, Bengaluru, Amrita Vishwa Vidyapeetham, Bengaluru, India
e-mail: bl.en.p2dsc20023@bl.students.amrita.edu

M. Supriya
e-mail: m_supriya@blr.amrita.edu

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_7

In order to cater to the needs of the society, the latest improvement to the cellular network generations is the fifth generation of wireless connection which has a bandwidth of the order of up to 10 GBit/s [1]. As the world is becoming more and more virtual and the devices become more and more smart these days, the network traffic has increased a lot. The increased data rates reduce the latency of the network, reduce the information loss, increase network availability, more reliability, higher network load handling capacity and more consistency of the network to the end-user. 5G is based on orthogonal frequency-division multiplexing (OFDM) modulation method that reduces interference between multiple channels [2]. The 5G technology uses the sub-6 GHz and the mmWave frequency spectrum. 5G is expected to be a driving factor for a 13.1 trillion dollars in the global economy. 5G technology is expected to have a major impact on the adoption and development of the latest technologies such as the virtual reality (VR), Internet of Things (IoT) and the artificial intelligence (AI). In IoT, it is expected to have a major impact on the edge computing technology. The current global air standard for the 5G technology is the 5G-NR (New Radio) developed by third Generation Partnership Project (3GPP), and the first version of specifications was released in 2017. It can handle the applications that are time consuming with the current telecommunication standard with ease and enhance the productivity.

The 5G technology is limited by the factors such as the inability to penetrate through the walls, as the frequency is high, the losses and the dissipation are also higher. This limits the range of 5G network within a short range, and its signal can also be affected even by the air. This range of 5G is much lesser than that of the 4G network. The initial costs of implementation are high, the current cellular devices are incompatible with the upgraded network, and this leads to an additional cost of upgrading to a new device to the consumers. Adaptive modulation and coding scheme (MCS) is used to ensure minimal data loss in 5G networks [3]. When the error rate falls below a threshold, the network operates on lower MCS to reduce the error rate. Here speed of the network is compromised in order to ensure minimal data loss.

This work focuses on modelling the 5G data based on the conventional and unconventional machine learning models to understand and predict the performance of the network under multiple conditions as a regression problem. The paper is structured as follows: Sect. 2 providing the literature survey; Sect. 3 describing the dataset and the features involved; Sect. 4 providing the exploratory data analysis; Sect. 5 providing the analysis and its results; Sect. 6 concludes the discussion.

2 Literature Review

A random forest (RF) regression model is implemented, and the performance of the model is estimated on the mean absolute error (MAE) and root mean square error (RMSE) in [4]. The best case obtained is a MAE of 163 with a RMSE value of 241, and the gradient boosted regressor (GBR) model implemented has given the best MAE of 100 with a RMSE of 154. Also, the history-based harmonic mean is modelled, and the implementation of an ordinary least squares (OLS) regression has generated a MAE of 231 and a RMSE of 340 on the throughput when analysed for a short-term prediction.

A logistic regression-based decision tree (DT) model for the accident injury severity which has shown a misclassification rate of 30%, and the cross-validation-based confusion matrix gives a misclassification rate of 32% in [5]. A DT model has achieved a 100% accuracy in condition monitoring of a milling tool in [6]. A RF model implementation for analysing pressure in suddenly expanded aerodynamic flows is applied in [7]. The analysis shows that the RF regression model has performed better than the K-means model for a nonlinear target variable prediction. Bayesian model proposed and implemented in [8] holds a better result with an $F1$-score of 0.785 while dealing with the data related to the consequences of construction accidents. The useful life prediction of lithium ion batteries by applying a adaptive extended kalman filter and genetic algorithm optimized support vector regression model has shown the best performance with the least MAE and RMSE in [9].

A support vector machine classifier (SVC) and linear discriminant analysis model has detected driver drowsiness in [10]. A SVC has been implemented in [11] for 2D indoor localization using RFID. A SVM-based heart disease detection system is modelled in [12]. A grinding wheel condition monitoring system using the acoustic characteristics is modelled in [13] applying CART and SVC. The extra trees (EXT) model has performed much better than the other models in the determination of bubble point pressure and the oil formation volume factor in [14]. The prediction of rock mass class on classification and regression tree-based AdaBoost model has achieved an $F1$-score of 0.77 and an accuracy of 0.865 in [15]. Gradient boosted regression (GBR) model is implemented for the degradation of prismatic cells, where the model has performed closer to the best model which is built upon the gradient boosted model in [16]. An extreme gradient boosted regressor (XGB) model on the interfacial tension between oil and injected gas has shown the least error in predicting the target variable with a R^2 value of 0.997 in [17]. A XGB model has been implemented in [18] to predict the sales of Big Mart, and the results show that it has outperformed the existing models. A construction cost prediction problem is modelled using a hybrid natural–light gradient boosted regression (LGB) model which has given an RMSE of 0.5 with a R^2 value of 0.99 in [19].

From the literature survey, it is noted that every model has performed the best on particular use case and the data is modelled using multiple models for a particular use case. In this work, multiple models are applied on the 5G dataset [20] as a regression problem, the analysis is performed, and the best model is identified.

3 Dataset Description

The dataset sourced from IEEE dataport site [20] contains about 68,118 records of data collected by means of 300 km of walking, 130 km of driving and 35 TB of data download. The data is collected over a loop area of 1300 m loop length. The dataset contains the following features such as the run number, sequence number, abstract signal strength, latitude, longitude, moving speed, compass direction, NR status/connection status, received signal strength indication (RSSI), reference signal received power (RSRP), reference signal received quality (RSRQ), reference signal signal-to-noise ratio (RSSNR), raw signal strength power (nrssRsrp), raw signal strength quality (nrssRsrq), raw signal strength signal-to-noise ratio (nrssSinr), throughput, mobility mode, trajectory direction and tower ID. All these data were collected using an Android API. Here the features such as the moving speed is recorded in m/s, the compass direction is measured in degrees, the connection status is a categorical variable with three options such as the connected, not restricted and none, mobility mode is also a categorical variable with two options such as walking and driving indicating the means of motion, and trajectory direction is also a categorical variable with two options such as clockwise or anticlockwise within the planned trajectory of motion and tower ID indicating to which tower the device is currently connected. The throughput is the target parameter in this work which is the rate of output obtained for a given condition at a given instance of time. All the other parameters indicate the signal qualities at a given run and sequence number. The run and sequence numbers are used for identification purposes. The data has been recorded over observing the throughput and the signal quality, the signal power with and without barriers between the carrier or the device and the tower to which it is connected. Here the accuracy of the data collected is subject to the accuracy and performance of the API.

4 The Exploratory Data Analysis

This section presents the exploratory data analysis of the 5G dataset described in the previous section with a focus on the throughput. Figure 1a, b indicates the throughput obtained along the contour traversed during the data collection process. It is observed that the throughput is minimal at some locations and high close to 5G speeds at other locations.

Figure 2 indicates the throughput obtained from each of the tower, and it could be noted that few towers are low in throughput, whereas others are close to the 5G spectrum range.

Figure 3a, b indicates the regression plot when the throughput is measured while driving a car and walking along the contour, respectively. The line represents the ordinary least square (OLS) regression line of the plot.

Figure 4 indicates the throughput obtained with the connection status a categorical variable which has three states which are the not restricted, connected and none, indicating the connection status with the 5G network.

Figure 5a, b indicates the throughput obtained when the traversal trajectory is clockwise and anticlockwise, respectively, in nature with the angle of motion as measured by the API, and the lines originating from the centre indicate the throughput

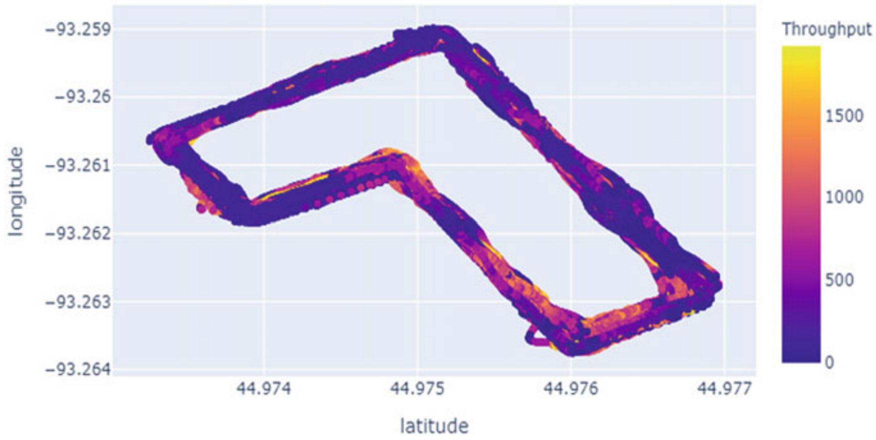


Fig. 1 Throughput obtained along the contour traversed

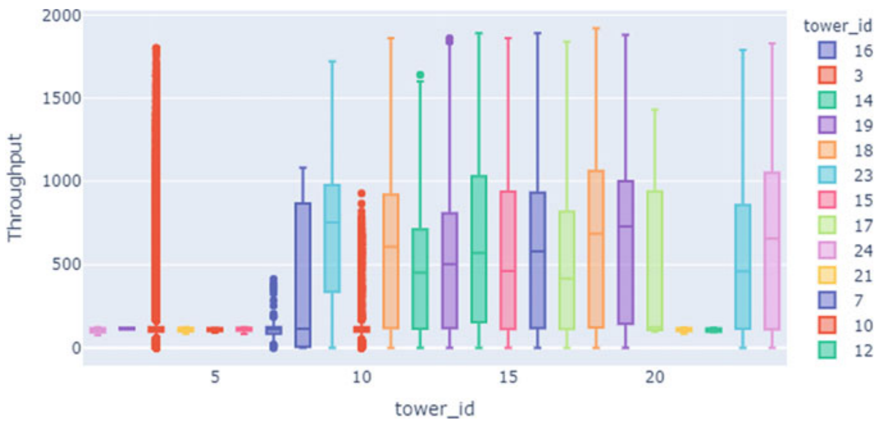


Fig. 2 Throughput obtained from the individual tower

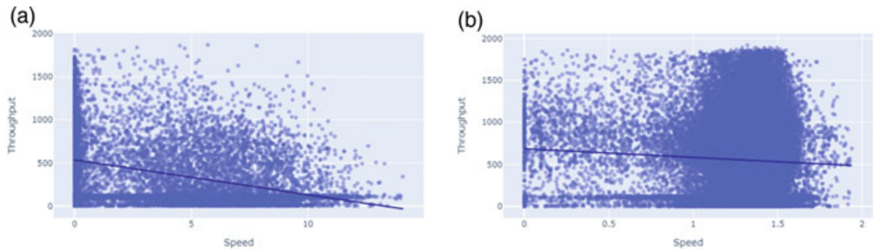


Fig. 3 Regression plot of throughput obtained **a** while driving a car and **b** while walking

Fig. 4 Throughput obtained with the connection status

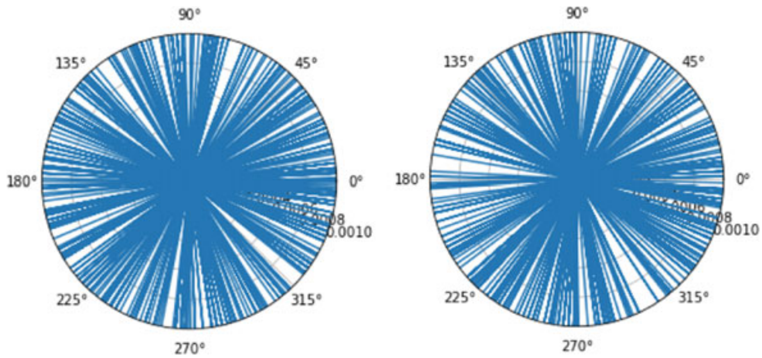


Fig. 5 Throughput obtained while direction of motion along the contour is clockwise and anti-clockwise with compass direction **a** as a polar plot when moving clockwise, **b** as a polar plot when moving anticlockwise

obtained at that particular angle. It is observed that the throughput is limited at some angles and in the 5G range at few other. Also, it has to be noted that there is significant difference in the throughputs obtained in case of the clockwise and anticlockwise directions of traversal. The accuracy of the plots is subject to the accuracy of the API. So, these categorical variables have been eliminated to reduce the modelling error, and the data is modelled as a regression problem with the other continuous variables.

5 Modelling and Analysis

The dataset is preprocessed to remove the null values detected, which are later replaced with 0. This preprocessing will enable the machine learning model to learn the dataset completely and present a model with higher accuracy. This in turn may reduce the errors and misclassification in the further process. Here the approach is to model the throughput selected as the target variable, and the data is modelled as a regression model not considering the categorical variables in the dataset. The categorical variables that are removed from the analysis are the abstractSignalStr, nrStatus, mobility mode, trajectory direction and tower ID. The less important features such as the run-num and seq-num are dropped from the analysis as they are just used for identification purposes. For modelling the data, the dataset is split into the standard train–test split ratio of 70:30. The models are applied onto the dataset as a regression problem.

$$\text{MAE} = \left(\frac{1}{n}\right)\sum_{i=1}^{n}|y_i - x_i| \tag{1}$$

$$\text{MSE} = \left(\frac{1}{n}\right)\sum_{i=1}^{n}(y_i - x_i)^2 \tag{2}$$

$$R^2 = 1 - (\text{SSR}/\text{SST}) \tag{3}$$

where

n	number of terms,
i	the ith term,
y_i	the actual value,
x_i	the calculated value,
MAE	mean absolute error,
MSE	mean squared error,
SSR	sum of squares of residuals,
SST	total sum of squares.

The parameters used for measuring the model performance are the mean absolute error (MAE), mean squared error (MSE) and the R-squared (R^2) values represented by Eqs. (1)–(3), respectively. MAE is the mean of the absolute difference between the actual and the predicted values. MSE is the mean of the square of the difference between the actual and the predicted value. R^2 is the measure of the correlation between the actual and the predicted values. SSR is the sum of the square of residues, and SST is the total sum of squares. Multiple models such as the decision tree (DT) model, random forest (RF) model, Naive Bayes (NB) model, support vector regressor (SVR) model, extra trees regressor (EXT) model, AdaBoost regressor (ABR) model, gradient boosted regressor (GBR) model, Xtreme gradient boosted regressor (XGB) model and light gradient boosted (LGB) model are applied on to the dataset. The performance of the DT model with different number of leaf nodes along with the tenfold cross-validated score of each of the case being analysed by varying the number of leaf nodes as 5, 50, 500 and 5000 is given in Table 1.

From Table 1, it could be noted that the model performance is increasing by increasing the number of leaf nodes of the decision tree model indicated by the reduction in the values of the parameters MAE and MSE. The performance of the models applied onto the 5G data which are the RF, NB, SVR, EXT, ABR, GBR, XGB and LGB is presented in Table 2. The performance of the models is measured using the parameters MAE, MSE and the R^2 values.

From the results in Table 2, EXT model has performed the best followed by the RF, DT model with 5000 leaf nodes, XGB and the LGB. The EXT model performance metrics are the MAE value of 140, MSE value of 46350 and the R^2 score of 0.783. As a validation of the performance of the models, cross-validation (CV) check is performed on the top five models, i.e. DT of 5000 leaf nodes, RF, EXT, LGB and XGB by dividing the dataset into ten partitions, i.e. a tenfold cross-validation is performed. The cross-validation scores of the top five models are given in Table 3.

From Table 3 of cross-validation scores, it is indicative that the EXT model has performed better than the other models applied onto the dataset.

Table 1 Performance of DT model

Number of leaf nodes in DT	MAE	MSE	R^2
5	256	125,144	0
50	232	106,202	0
500	201	85,083	0
5000	173	81,207	0

Table 2 Performance of other models

Algorithm	MAE	MSE	R^2
RF	145	47,746	0.777
NB	474	305,176	−0.425
SVR	391	219,580	−0.025
EXT	140	46,350	0.783
ABR	261	118,769	0.445
GBR	229	100,120	0.532
XGB	179	63,981	0.701
LGB	194	74,407	0.652

Table 3 Cross-validation scores of top five models

CV score (tenfold)	DT	RF	EXT	LGB	XGB
1	−0.1797	0.3815	0.3977	0.4256	0.3386
2	0.0467	0.489	0.5026	0.499	0.4775
3	0.1057	0.5312	0.5299	0.531	0.4932
4	−0.1097	0.4481	0.4638	0.4475	0.3604
5	0.1051	0.4217	0.4239	0.4359	0.3904
6	0.0604	0.4703	0.4794	0.5181	0.5189
7	−0.1729	0.4312	0.4593	0.4259	0.3684
8	−0.0044	0.4288	0.4385	0.4664	0.4365
9	0.0016	0.5046	0.5288	0.4514	0.4492
10	−0.0231	0.3854	0.3868	0.4331	0.4188

6 Conclusion

5G is one of the emerging technologies globally which is finding use case in almost every application and the successor to the current 4G technology. In this paper, the throughput obtained under various conditions is modelled as a regression problem eliminating the categorical variables. From the analysis, it is observed that, among the models being analysed, the EXT model has performed the best with a MAE of 140, MSE of 46350 and a R^2 score of 0.783, and the results are in accordance with the results of the tenfold cross-validated score. The future scope of work is to apply other machine learning models upon the data and identify the best model that can be applied onto 5G dataset as a regression problem.

References

1. https://www.qualcomm.com/5g/what-is-5g
2. https://www.qualcomm.com/media/documents/files/5g-research-on-waveform-and-multiple-access-techniques.pdf
3. https://www.3gpp.org//ftp//tsg_ran//WG1_RL1//TSGR1_17//docs//PDFs//R1-00-1395.pdf
4. Narayanan A, Ramadan E, Mehta R, Hu X, Liu Q, Fezeu RAK, Dayalan UK, Verma S, Ji P, Li T, Qian F, Zhang Z-L (2020) LUMOS5G: mapping and predicting commercial mmWave 5G throughput. In: Proceedings of the ACM internet measurement conference, IMC'20. Association for Computing Machinery, New York, NY, USA, pp 176–193
5. Rezapour M, Molan AM, Ksaibati K (2020) Analyzing injury severity of motorcycle at-fault crashes using machine learning techniques, decision tree and logistic regression models. Int J Transp Sci Technol 9(2):89–99
6. Mohanraj T, Yerchuru J, Krishnan H, Nithin Aravind RS, Yameni R (2021) Development of tool condition monitoring system in end milling process using wavelet features and hoelder's exponent with machine learning algorithms. Measur J Int Measur Confed 173

7. Afzal A, Aabid A, Khan A, Khan SA, Rajak U, Verma TN, Kumar R (2020) Response surface analysis, clustering, and random forest regression of pressure in suddenly expanded high-speed aerodynamic flows. Aerosp Sci Technol 107:106318
8. Zhu R, Hu X, Hou J, Li X (2021) Application of machine learning techniques for predicting the consequences of construction accidents in China. Process Safety Environ Protection 145:293–302
9. Xue Z, Zhang Y, Cheng C, Ma G (2020) Remaining useful life prediction of lithium-ion batteries with adaptive unscented kalman filter and optimized support vector regression. Neurocomputing 376:95–102
10. Madireddy R, Anudeep DSK, Poorna SS, Anuraj K, Gokul Krishna M, Balaji A, Venkat DJ (2021) Driver drowsiness detection system using conventional machine learning. Lecture notes on data engineering and communications technologies 58:407–415
11. Aravind Raamasamy S, Shanmuga Pradeep P, Mani Madhav Goud CH, Viswanathan Babu CA, Jayakumar M (2021) Analysis of machine learning algorithms for RFID based 2D indoor localization. Lecture notes on data engineering and communications technologies 58:229–242
12. Anusha M, Suresh K, Chandana M (2021) Earlier prediction on the heart disease based on supervised machine learning techniques. In: Proceedings—5th international conference on intelligent computing and control systems, ICICCS 2021, pp 1696–1703
13. Rameshkumar K, Mouli DSB, Shivith K (2021) Machine learning models for predicting grinding wheel conditions using acoustic emission features. SAE Int J Mater Manuf 14(4)
14. Seyyedattar M, Ghiasi MM, Zendehboudi S, Butt S (2020) Determination of bubble point pressure and oil formation volume factor: extra trees compared with LSSVM-CSA hybrid and ANFIS models. Fuel 269:116834
15. Liu Q, Wang X, Huang X, Yin X (2020) Prediction model of rock mass class using classification and regression tree integrated adaboost algorithm based on tbm driving data. Tunnell Undergr Space Technol 106:103595
16. Wang F-K, Mamo T (2020) Gradient boosted regression model for the degradation analysis of prismatic cells. Comput Indust Eng 144:106494
17. Zhang J, Sun Y, Shang L, Feng Q, Gong L, Kuankuan W (2020) A unified intelligent model for estimating the (gas + n-alkane) interfacial tension based on the extreme gradient boosting (XGBoost) trees. Fuel 282:118783
18. Ranjitha P, Spandana M (2021) Predictive analysis for big mart sales using machine learning algorithms. In: Proceedings—5th international conference on intelligent computing and control systems, ICICCS 2021, pp 1416–1421
19. Chakraborty D, Elhegazy H, Elzarka H, Gutierrez L (2020) A novel construction cost prediction model using hybrid natural and light gradient boosting. Adv Eng Inform 46:101201
20. https://ieee-dataport.org/open-access/lumos5g-dataset

A Novel Technique to Detect Inappropriate Content Accessed by Children on Smartphone

Savita Yadav, Pinaki Chakraborty, Prabhat Mittal, Aditya Kumar, and Harshit Gupta

Abstract The increased access to the Internet on smartphone has enhanced the possibility of exposure of children to content inappropriate for them. A smartphone app to automatically record and analyze inappropriate online material was developed and provided to children of three age groups, viz. four to six, seven and eight and nine and ten years. The smartphone app determined the time spent using smartphones, and the number of times adult text, adult graphics, violent text and violent graphics was accessed by children in a week on an average. One-way multivariate analysis of variance was used to find out significant differences in the five parameters among the three age groups. Results showed that children aged nine and ten years spent 193.37 min on smartphones and accessed content comprising of adult graphics and violent graphics on 3.93 and 30.63 times per week on an average. Younger children aged four to six and seven and eight years were found to use smartphones less with 128.37 and 151.17 min on an average, respectively, and get exposed to inappropriate content less frequently. Children in all the age groups got exposed to inappropriate graphical content more frequently as compared to inappropriate textual content. The app facilitates awareness of parents about the online activities of their children. Timely intervention of parents may prepare children to counteract unpleasant online

S. Yadav (✉) · P. Chakraborty · A. Kumar · H. Gupta
Department of Computer Science and Engineering, Netaji Subhas University of Technology, New Delhi, India
e-mail: savitaydv@yahoo.com

P. Chakraborty
e-mail: pinaki_chakraborty_163@yahoo.com

A. Kumar
e-mail: aditya.kumar.co19@nsut.ac.in

H. Gupta
e-mail: harshit.co19@nsut.ac.in

P. Mittal
Department of Commerce, Satyawati College (Evening), University of Delhi, Delhi, India
e-mail: profmittal@yahoo.co.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_8

experiences. This cognizance of children will thus empower them to benefit from the abundant wealth of information available online via smartphones.

Keywords Smartphone · Internet · Children · Adult content · Violent content

1 Introduction

Smartphones are used by children of all age groups [1]. Children surf the Internet [2] and are initiated into downloading the software applications on smartphones at a young age [3]. With this early exposure to the opportunities available online, they are also exposed to the risks and hazards of the online digital world [4, 5]. Innumerable smartphone apps along with abundant digital content specifically directed toward children are available on the Internet [6]. The utility as well as the advantages of countless child-oriented software has been of much interest as well as subject of debate in the research community. The content of a number of smartphone apps for children is often found to be objectionable [7]. Children are regularly exposed to unsuitable content on YouTube channels as well [8]. Parents are in a unique situation and find themselves indecisive on whether to allow their children to download the smartphone apps of their choice [9]. Many times the offline experiences of parents help them in selecting educational apps for their children irrespective of the interface and user experience afforded by the smartphone app [10]. In addition, it is often observed that the parents consider the description and the feedback about the content of the smartphone apps from different sources while selecting useful smartphone apps for their children [11]. However, the maturity ratings provided to a smartphone app may vary across different platforms and hence are inaccurate as well as inconsistent [12]. It is frequently observed that even the smartphone apps thus selected expose children to inappropriate content. Many of the smartphone apps for children contain links to unsuitable websites and in-app advertisements that may be harmful for them [13]. Further, children may be encouraged to make in-app purchases [14] in order to make use of the advance features of the app or to continue using the smartphone app in future. Parents have to exercise caution and have to be extremely proactive so as to make an informed choice regarding useful smartphone apps for their children. However, this becomes challenging at times for parents, and they find it difficult to monitor online activities of their children [15].

Children use smartphones and other digital devices mostly for academic purpose and watching videos [16]. Children are inquisitive and are often led by their search for answers. Internet is popular among them in their pursuit of gathering knowledge on the topics of their interest. In this quest, at times, children come in close interaction with strangers online. Social media websites uses are gaining immense popularity among children [17]. As children grow, they create their own user profiles on social networking sites. It has been found that children fabricate their exact age [18] and manage to override the minimum age requirement conditions for having a valid online user ID. This misdemeanor may lead to harassment of children online in the hands of bullies. Children may be intimidated online by a bully or a group with threatening emails, rude text messages and alteration in their profiles with inappropriate images. Such an incident is colloquially known as cyber bullying [19]. Children are vulnerable and may be manipulated into misconduct by bullies. Cyber bullying has been found to be leading to depression among children [20] and is known to have negative psychological consequences [21], which can have a lifelong effect on the personality of children. Children, due to fear of being reprimanded, may not even disclose such happenings to their parents [22]. Overall it may pose a grave threat to the well-being of the children. Hence, it is important that healthcare workers regularly screen for bullying among children in order to prevent the harms posed by cyber bullying [23]. In addition, initiatives by schools and policy makers to prevent traditional bullying should also accommodate provisions to confront cyber bullying [19, 24].

There is a need to balance the concern of parents to monitor the activities of their children on smartphones with the desire of children to be independent in their choice [15]. The objective of the paper is to develop a smartphone app to capture and filter the inappropriate content accessed online by children of different age groups. The information attained by this analysis may then be used by parents to advise and counsel their children regarding precautions to be undertaken by children while engaging in online activities.

2 Related Work

2.1 Apprehensions of Parents and Needs of Children

The excessive use of smartphone gaming and social networking sites can lead to smartphone addiction among fifth-grade children [25]. With children spending significant amount of time online, they are also more prone to cyber bullying [26]. The restrictive mediation approach by parents can reduce the risks associated with smartphone addiction [25]. However, often the use of parental control leads to a constrained relationship between parents and children [27]. In addition, the strategies employed for online security of children often provide safe platform but at the same time reduce the number of available opportunities [28]. It is opined that the analysis of the reviews on parental control apps can be used to improve their design and thereby provide an enhanced user experience to the parents and children alike [27]. The design of the parental control apps should not be overtly overbearing but rather provide for a technical intervention that supports the needs of parents as well as children [29].

2.2 Applications to Protect Interests of Children

Children use mobile devices and are thus often subjected to content and privacy risks [30]. To safeguard children from these two risks, an automatic system to detect improper content and a user interface to make parents aware of the privacy risks can be designed [30]. Further, an application to raise awareness among children and adults about the conception of harmful digital content and online safety can be built [5]. A model to automatically predict the maturity-level rating of a mobile app with high accuracy and low cost can be developed [12]. Researchers have also planned and developed sophisticated parental control system which is effective in preventing unintended denial as well as unintentional access to sensitive processes, at the same time provide many features like location tracking, emergency notifications and many more [31]. A framework to automatically inspect the contents of an android app for inappropriate material, for children below 12 years of age, has been developed by Luo et al. [7]. Filtering children's videos containing sparse inappropriate content has also been achieved [32]. Mobile service which limits the use of mobile by considering it as a family activity has been found to be effective in developing congenial atmosphere for parental mediation [15]. With many online platforms like YouTube kids, child-directed contents are being shared which at many times are unsuitable for them [33, 34]. A deep learning design to raise a flag and report the same has also been developed [33]. It is even possible to detect and identify obscene contents from the text posted on various sites using natural language processing and thus save children from unpleasant experiences [2].

2.3 Challenges

YouTube videos are increasingly being watched by children as young as three years which may raise the concern of exposure of children to disturbing videos [35]. More extensive research may further open avenues to better understand the extent of harmful exposure children are prone to when watching YouTube videos and how to prevent the same [8]. The need to protect children from vices of inappropriate content and cyber bullying has steered many researchers into developing content filters for smartphone apps and suitable measures to report cyber bullying of children. However, the reach and popularity of such filters and cyber bullying countermeasures to protect children online are yet to be determined. Several of these filters are intrusive [15], too complex and have many drawbacks in their approach to render them ineffective [2]. Many of the parents may not even understand how to install the available software and their underlying functioning. Children may even resist if they are unnecessarily denied access to worthy, informative and age appropriate content [36]. Parents need to be cautious of the actions of their children once they are initiated into Internet surfing [37] and act accordingly. It has been found that some child-directed videos available online contain objectionable and detrimental content for children [35] and demand immediate attention. In addition, parents have to also ensure that their children do not join social networking sites before the age of 13 years [38].

3 Materials and Methods

3.1 The Smartphone App

We developed a smartphone app to detect inappropriate content accessed by the children on the Internet using a smartphone. The app can identify four types of inappropriate content, viz. adult text, adult graphics, violent text and violent graphics.

- *Text-Based Classification*: We used a combination of *k*-nearest neighbors and random forests techniques for classifying the text accessed by children on their smartphones. Google's Cloud Vision API was used to extract text from the screen captures of the smartphones. This API extracted text from a given image and classified whether the text was inappropriate for the child or not (Fig. 1).The smartphone app searched for keywords signifying vulgar, obscene, swearing and their synonyms before classifying the text as adult text. The content was put into violent text category if it included words like hit, weapon, kill, suicide, torture, bully and their synonyms.

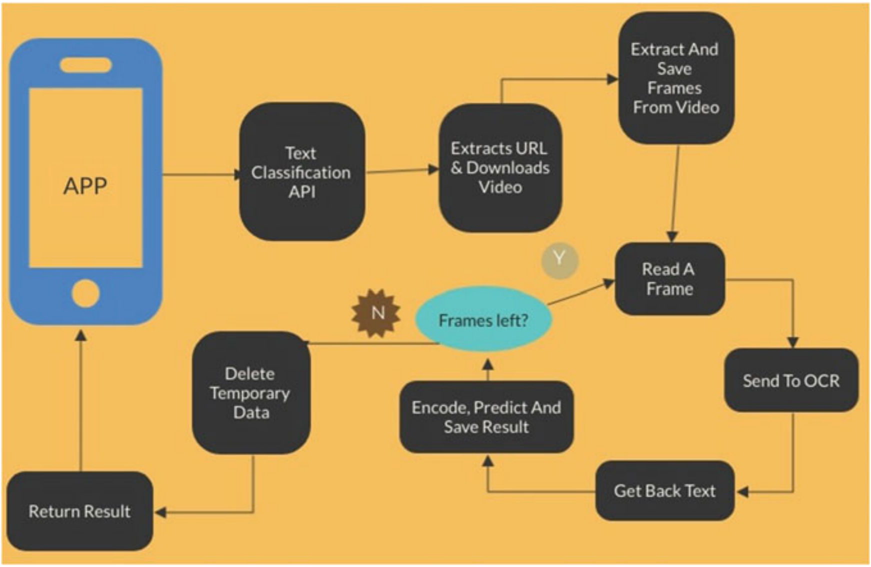


Fig. 1 Detection of inappropriate text

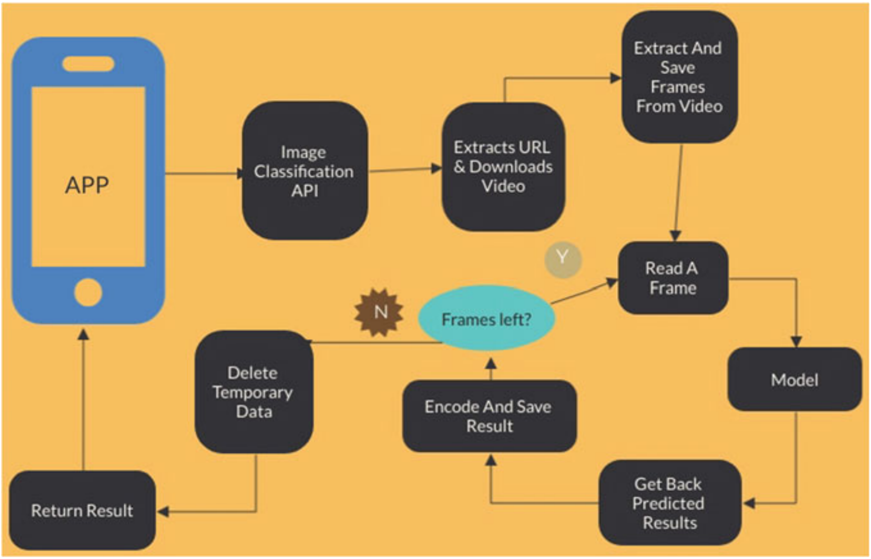


Fig. 2 Detection of inappropriate images

- *Image-Based Classification*: The image-based classification used convolutional neural networks and transfer learning on Inception v3 to classify the results according to the weights of ImageNet and further trained on labeled custom data (Fig. 2). The training data set consisted of explicit graphics showing obscene acts were earmarked as adult graphics. Further, the content showing racing of vehicles, fight scenes and blood were classified as violent graphics.

The app generated weekly reports on the basis of the collected and analyzed data for parents to examine (Fig. 3).

3.2 Experimental Protocol

Parents of children aged four to ten years were reached through emails and instant messages to participate in the study. We selected 90 agreeing parents to take part in the study. The willing parents were asked to install the smartphone app on the mobile devices used by their respective child. The smartphone app worked in two modes, viz. parent mode and child mode. The app ran as a background process and recorded the time spent as well as content viewed on the smartphone in the child mode only. Parents were asked to switch the app to the child mode before giving their smartphone to the child participating in the study for a period of one week. After the time span of one week, parents were asked to check the weekly report of their child stating the average time spent using a smartphone and the average number of

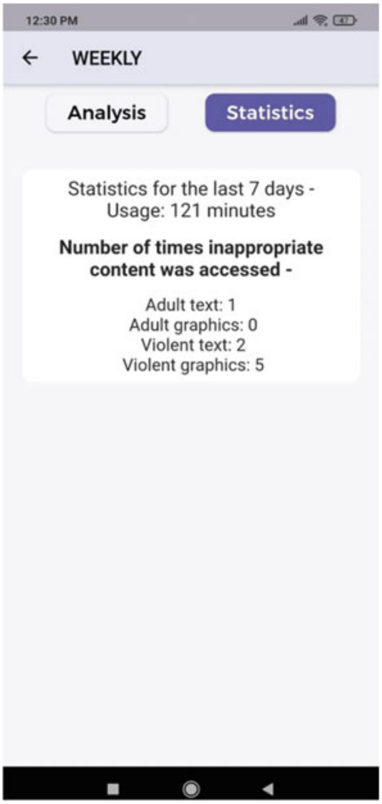


Fig. 3 Screenshot of the app

times inappropriate content, viz. adult text, adult graphics, violent text and violent graphics which were accessed. The children were divided into three categories of age, viz. four to six, seven and eight and nine and ten years for analysis of the results. Each age group comprised of usage statistics of 30 children.

3.3 Statistical Analysis

One-way multivariate analysis of variance (MANOVA) was used to find out significant differences in the five parameters, viz. average time spent using smartphone (A), average number of times adult text was accessed (B), average number of times adult graphics was accessed (C), average number of times violent text was accessed (D) and average number of times violent graphics was accessed (E) among the three age groups. To determine how the dependent parameters differ for the independent variable, we used the tests of between-subjects effects. Further, we carried out post hoc analysis using Tuckey-HSD and Games-Howell test to find the differences in the most significant age group. We conducted the Levene's test of equality of error variances to test the homogeneity of variances in groups.

4 Results

Children aged four to six years spent on an average 128.37 min using smartphone per week which increased to 151.17 min for children aged seven and eight years and was found to be maximum for the children in the oldest age group at 193.37 min (Table 1). It was found that the average number of times adult text was accessed for children in the youngest two age groups, viz. four to six and seven and eight years, which were nil. However, nine- and ten-year-old children were found to access adult text on an average 1.10 times per week. Children aged four to six and seven and eight years were found to access adult graphics on an average 0.17 and 0.40 times per week, respectively, which escalated to 3.93 for the nine- and ten-year-old children. The access of violent text was more than the adult text for each of the age groups and was found to be 0.30, 1.90 and 4.13 times per week for the three age groups, respectively. It was evident that the children preferred to access violent graphics more than violent text. The average number of times violent graphics was accessed by the three age groups in the specified order was 5.17, 10.57 and 30.63 times per week.

The P-value was found to be 0.000 for Wilks' lambda for the different age groups (Table 2). It was found that the five parameters, viz. average time spent using smartphone (A), average number of times adult text was accessed (B), average number of times adult graphics was accessed (C), average number of times violent text was accessed (D) and average number of times violent graphics was accessed (E) are significantly dependent on age ($F(10,166) = 20.685$, $P < 0.001$, Wilk's lambda = 0.198, partial $\eta^2 = 0.555$).

Table 3 shows that age group has a statistically significant effect on all the parameters ($P < 0.001$). An η^2 of 0.241 of average time spent using smartphone (A) indicates that 24.1% of the total variance is explained after accounting for variance

Table 1 Average time spent by children using smartphone and average number of times inappropriate content was accessed

Age group	Number of children	Average time spent using smartphone (minutes) (A)	Average number of times inappropriate content was accessed			
			Adult text (B)	Adult graphics (C)	Violent text (D)	Violent graphics (E)
4–6 years	30	128.37 ± 11.44	0.00 ± 0.00	0.17 ± 0.08	0.30 ± 0.15	5.17 ± 1.12
7 and 8 years	30	151.17 ± 4.29	0.00 ± 0.00	0.40 ± 0.11	1.90 ± 0.35	10.57 ± 1.33
9 and 10 years	30	193.37 ± 9.32	1.10 ± 0.31	3.93 ± 0.63	4.13 ± 0.62	30.63 ± 2.12

Table 2 Results of MANOVA

Effect		Value	F	Hypothesis df	Error df	P-value	Partial η^2
Intercept	Pillai's trace	0.952	330.333[a]	5.000	83.000	0.000	0.952
	Wilks' lambda	0.048	330.333[a]	5.000	83.000	0.000	0.952
	Hotelling's trace	19.900	330.333[a]	5.000	83.000	0.000	0.952
	Roy's largest root	19.900	330.333[a]	5.000	83.000	0.000	0.952
Age group	Pillai's trace	0.865	12.810	10.000	168.000	0.000	0.433
	Wilks' lambda	0.198	20.685[a]	10.000	166.000	0.000	0.555
	Hotelling's trace	3.725	30.543	10.000	164.000	0.000	0.651
	Roy's largest root	3.637	61.097[c]	5.000	84.000	0.000	0.784

explained by other parameters in the model. The highest η^2 of average number of times violent graphics was accessed (E) which indicates the maximum effect (62.3% of total variance) among all the parameters.

We tested the null hypothesis that the error variance of the dependent parameter is equal across groups (Table 4). Here, a. Design: Intercept + Age Group. The results confirmed that the age groups are not homogenous in all parameters except for average time spent using smartphone (A). We carried out post hoc analysis using Tuckey-HSD for finding the differences in the age groups for average time spent using smartphone (A).

The results of post hoc analysis in Table 5 show low significance value ($P < 0.01$) which confirms that there is significant difference among parameters in all the age groups except for average time spent using smartphone (A) and average number of times adult graphics was accessed (C). It was found that for the age groups four to six and seven and eight years, the difference in average time spent using smartphone (A) and average number of times adult graphics was accessed (C) was insignificant ($P > 0.05$).

5 Discussion

Children aged four to six years are quite young to be searching on the Internet on their own. They are still learning the skills to operate a smartphone all by themselves. They generally perform voice search using minimum keywords to find their favorite videos, and this at times results in the popping of unsuitable content. The lack of awareness often misleads these children to click on coarse content. Children at this age were

Table 3 Effect of age group on average time spent by children using smartphone and average number of times inappropriate content was accessed

Source	Dependent parameter	Type III sum of squares	df	Mean square	F	P-value	Partial η^2
Corrected model	A	65,256.800[a]	2	32,628.400	13.816	0.000	0.241
	B	24.200[c]	2	12.100	12.729	0.000	0.226
	C	267.267[d]	2	133.633	31.659	0.000	0.421
	D	222.422[e]	2	111.211	20.831	0.000	0.324
	E	10,803.822[f]	2	5401.911	71.921	0.000	0.623
Intercept	A	2,236,344.100	1	2,236,344.100	946.920	0.000	0.916
	B	12.100	1	12.100	12.729	0.000	0.128
	C	202.500	1	202.500	47.974	0.000	0.355
	D	401.111	1	401.111	75.133	0.000	0.463
	E	21,498.678	1	21,498.678	286.232	0.000	0.767
Age group	A	65,256.800	2	32,628.400	13.816	0.000	0.241
	B	24.200	2	12.100	12.729	0.000	0.226
	C	267.267	2	133.633	31.659	0.000	0.421
	D	222.422	2	111.211	20.831	0.000	0.324
	E	10,803.822	2	5401.911	71.921	0.000	0.623
Error	A	205,468.100	87	2361.702			
	B	82.700	87	0.951			
	C	367.233	87	4.221			
	D	464.467	87	5.339			
	E	6534.500	87	75.109			
Total	A	2,507,069.000	90				
	B	119.000	90				
	C	837.000	90				
	D	1088.000	90				
	E	38,837.000	90				
Corrected total	A	270,724.900	89				
	B	106.900	89				
	C	634.500	89				
	D	686.889	89				
	E	17,338.322	89				

found to mostly search for their favorite toys, games and cartoons. It was found that the appearance of in-app advertisement and notifications of related apps while watching their favorite videos would make these children unintentionally click on the same. As a result, children watched graphical obscene content on a few occasions. In addition, children were also found watching violent visuals when randomly clicking

Table 4 Levene's test of equality of error variances[a]

Dependent variable	F	df1	df2	P-value
A	2.049	2	87	0.135
B	27.042	2	87	0.000
C	56.837	2	87	0.000
D	21.469	2	87	0.000
E	5.968	2	87	0.004

and accessing any of the popped up links. This would expose these young children to adult and violent content unknowingly which could be harmful for them. Parents should take appropriate measures and advise children in this age group to refrain from clicking on any random sites and ensure that children make use of smartphones in the presence of an adult only.

Children aged seven and eight years gain a better understanding of smartphones and their applications. They spend considerably more time on smartphones searching for the games and videos of their choice as compared to the youngest age group. These children mostly like to search for racing game videos, action videos and sports videos which also contain advertisement not suitable for children. At this age, children gradually develop an awareness of vast number of websites and portals disseminating children's videos. However, they have not yet matured to understand the risks involved while surfing through such sites. It was observed that though these children did not specifically search for adult and violent content but were exposed to such sites through the pop-up advertisements. The exposure to adult content was much lesser in comparison with the violent content. This was attributed to the preference of the children in this age group for the violent videos and the appearance of related videos on the screen. These children would often click on the suggested violent videos and preferred to watch them. The exposure to gruesome violent videos may have a long-term impact on the behavior of these children. The intervention of parents at appropriate stage will benefit the children and steer away them from such dangerous content.

Children aged nine and ten years are confident and have a sound knowledge of the functionalities of smartphones. They are able to use almost all the applications of smartphones and spend maximum time using smartphones among the three age groups. These children like to explore Internet using smartphones on their own and are prone to be subjected to online hazards. These children like to imitate adults and boast of their accomplishments. While searching Internet for random purposes, they would intentionally click on links which directed them to objectionable videos like crime videos and obscene videos. They would generally prefer to watch more videos with graphics rather than reading text to satisfy their curiosity. This could prove to be harmful for the psychological development of the children and pose grave threat to their overall well-being. Parents have to be extremely cautious with children in this age group. A balanced approach comprising of counseling with subtle warnings

Table 5 Results of post hoc analysis

Dependent variable		(I) Age group	(J) Age group	Mean difference (I-J)	Standard error	P-value
A	Tuckey-HSD	4–6	7–8	−22.80	12.548	0.170
			9–10	−65.00*	12.548	0.000
		7–8	4–6	22.80	12.548	0.170
			9–10	−42.20*	12.548	0.003
		9–10	4–6	65.00*	12.548	0.000
			7–8	42.20*	12.548	0.003
B	Games-Howell	4–6	7–8	0.00	0.000	-
			9–10	−1.10*	0.308	0.004
		7–8	4–6	0.00	0.000	-
			9–10	−1.10*	0.308	0.004
		9–10	4–6	1.10*	0.308	0.004
			7–8	1.10*	0.308	0.004
C	Games-Howell	4–6	7–8	−0.23	0.141	0.233
			9–10	−3.77*	0.640	0.000
		7–8	4–6	0.23	0.141	0.233
			9–10	−3.53*	0.644	0.000
		9–10	4–6	3.77*	0.640	0.000
			7–8	3.53*	0.644	0.000
D	Games-Howell	4–6	7–8	−1.60*	0.385	0.000
			9–10	−3.83*	0.639	0.000
		7–8	4–6	1.60*	0.385	0.000
			9–10	−2.23*	0.715	0.008
		9–10	4–6	3.83*	0.639	0.000
			7–8	2.23*	0.715	0.008
E	Games-Howell	4–6	7–8	−5.40*	1.737	0.008
			9–10	−25.47*	2.398	0.000
		7–8	4–6	5.40*	1.737	0.008
			9–10	−20.07*	2.501	0.000
		9–10	4–6	25.47*	2.398	0.000
			7–8	20.07*	2.501	0.000

and suitable rewards for children in this age group will enlighten them with dos and don’ts while surfing the Internet on smartphones.

In this study, all the children in different age groups belonged to upper-middle class background and were residents of New Delhi. A more elaborate study comprising of large sample spread across diverse social and economic strata will provide a detailed understanding of the Internet search behavior of children.

Children in almost all the age groups access Internet via smartphones. The contributions of the present study were threefold as follows.

- A smartphone app was developed to collect statistics related to the time spent and type of content accessed by children while they are online.
- The type of content searched and preferred by children of three different age groups was studied and analyzed.
- The simple app allowed parents to conveniently monitor the online searches of their children and proactively take decisive measures for their online safety.

6 Conclusion

Smartphones are popular among children and list in their most liked digital device. With increase in age, children spend more time using smartphones. Children are not only curious to find answers to their queries but also like to watch online videos of their choice on smartphones. It is found that in their quest to seek information and get entertained using smartphones, they are many times exposed to harmful content as well. They are far more vulnerable to be exposed to inappropriate content on smartphones as they grow older. We developed an app that records and analyzes the content searched and viewed by children for later analysis by the parents. The awareness of parents about the online activities of their children will ensure proper guidance and counseling of children by them. This ensures that children learn to make responsible decision when online to safeguard their interests.

References

1. Kabali HK, Irigoyen MM, Nunez-Davis R, Budacki JG, Mohanty SH, Leister KP, Bonner RL (2015) Exposure and use of mobile media devices by young children. Pediatrics 136(6):1044–1050
2. Barrientos GM, Alaiz-Rodríguez R, González-Castro V, Parnell AC (2020) Machine learning techniques for the detection of inappropriate erotic content in text. Int J Comput Intell Syst 13(1):591–603
3. Yadav S, Chakraborty P (2018) Using smartphones with suitable apps can be safe and even useful if they are not misused or overused. Acta Paediatr 107(3):384–387
4. Wold T, Aristodemou E, Dunkels E, Laouris Y (2009) Inappropriate content. In: Livingstone S, Haddon L (eds) Kids online: opportunities and risks for children. Policy Press, pp 135–146
5. Poblet M, Teodoro E, González-Conejero J, Varela R, Casanovas P (2017) A co-regulatory approach to stay safe online: reporting inappropriate content with the MediaKids mobile app. J Family Stud 23(2):180–197
6. Goodwin K, Highfield K (2012) iTouch and iLearn: an examination of "educational" apps. In: Presented in early education and technology for children conference
7. Luo Q, Liu J, Wang J, Tan Y, Cao Y, Kato N (2020) Automatic content inspection and forensics for children android apps. IEEE Internet Things J 7(8):7123–7134

8. Alshamrani S (2020) Detecting and measuring the exposure of children and adolescents to inappropriate comments in YouTube. In: Proceedings of the twenty-ninth ACM international conference on information and knowledge management (pp 3213–3216)
9. Leathers H, Summers P, Desollar A (2013) Toddlers on technology: a parents' guide. Author House
10. Brito R, Dias P (2020) Which apps are good for my children?: how the parents of young children select apps. Int J Child-Comput Interact 26:100188
11. Papadakis S, Kalogiannakis M (2017) Mobile educational applications for children: what educators and parents need to know. Int J Mobile Learn Organ 11(3):256–277
12. Hu B, Liu B, Gong NZ, Kong D, Jin H (2015) Protecting your children from inappropriate content in mobile apps: an automatic maturity rating framework. In: Proceedings of the twenty-fourth ACM international on conference on information and knowledge management, (pp 1111–1120).
13. Chen Y, Zhu S, Xu H, Zhou Y (2013) Children's exposure to mobile in-app advertising: an analysis of content appropriateness. In: Proceedings of the international conference on social computing, (pp 196–203)
14. Radesky J, Chassiakos YL, Ameenuddin N, Navsaria D (2020) Digital advertising to children. Pediatrics 146(1):e20201681
15. Ko M, Choi S, Yang S, Lee J, Lee U (2015) FamiLync: facilitating participatory parental mediation of adolescents' smartphone use. In: Proceedings of the ACM international joint conference on pervasive and ubiquitous computing, (pp 867–878)
16. Howie EK, McNally S, Straker LM (2020) Exploring the reliability and validity of the TechU-Q to evaluate device and purpose specific screen use in preschool children and parents. J Child Fam Stud 29(10):2879–2889
17. O'Keeffe GS, Clarke-Pearson K (2011) The impact of social media on children, adolescents, and families. Pediatr 127(4):800–804
18. Park SH, Han SP, Huh SY, Lee H (2009) Preprocessing uncertain user profile data: Inferring user's actual age from ages of the user's neighbors. In: Proceedings of the IEEE twenty-fifth international conference on data engineering, (pp 1619–1624)
19. Snakenborg J, Van Acker R, Gable RA (2011) Cyberbullying: prevention and intervention to protect our children and youth. Preventing School Fail: Altern Educ Child Youth 55(2):88–95
20. Hamm MP, Newton AS, Chisholm A, Shulhan J, Milne A, Sundar P, Ennis H, Scott SD, Hartling L (2015) Prevalence and effect of cyberbullying on children and young people: a scoping review of social media studies. JAMA Pediatr 169(8):770–777
21. Kwan I, Dickson K, Richardson M, MacDowall W, Burchett H, Stansfield C, Brunton G, Sutcliffe K, Thomas J (2020) Cyberbullying and children and young people's mental health: a systematic map of systematic reviews. Cyberpsychol Behav Soc Netw 23(2):72–82
22. Mishna F, Saini M, Solomon S (2009) Ongoing and online: children and youth's perceptions of cyber bullying. Child Youth Serv Rev 31(12):1222–1228
23. Vaillancourt T, Faris R, Mishna F (2017) Cyberbullying in children and youth: implications for health and clinical practice. Can J Psychiatry 62(6):368–373
24. John A, Glendenning AC, Marchant A, Montgomery P, Stewart A, Wood S, Lloyd K, Hawton K (2018) Self-harm, suicidal behaviours, and cyberbullying in children and young people: systematic review. J Med Int Res 20(4):e129
25. Chang FC, Chiu CH, Chen PH, Chiang JT, Miao NF, Chuang HY, Liu S (2019) Children's use of mobile devices, smartphone addiction and parental mediation in Taiwan. Comput Hum Behav 93:25–32
26. van Tiel J (2020) Cyberbullying, an overlooked and ever growing danger to the development of children. Technical report, KidsRights
27. Alelyani T, Ghosh AK, Moralez L, Guha S, Wisniewski P (2019) Examining parent versus child reviews of parental control apps on Google play. In: Proceedings of the international conference on human-computer interaction, pp 3–21
28. Livingstone S, Ólafsson K, Helsper EJ, Lupiáñez-Villanueva F, Veltri GA, Folkvord F (2017) Maximizing opportunities and minimizing risks for children online: the role of digital skills in emerging strategies of parental mediation. J Commun 67(1):82–105

29. Zaman B, Nouwen M (2016) Parental controls: advice for parents, researchers and industry. Technical report, EU Kids Online
30. Chen Y (2014) Protect children online safety on social media and mobile platforms. Doctoral dissertation, Pennsylvania State University
31. Tack J (2016) Protect my child: a new paradigm in parental control. Master's dissertation, Pennsylvania State University
32. Han W, Ansingkar M (2020) Discovery of elsagate: detection of sparse inappropriate content from kids videos. In: Proceedings of the zooming innovation in consumer technologies conference, (pp 46–47)
33. Tahir R, Ahmed F, Saeed H, Ali S, Zaffar F, Wilson C (2019) Bringing the kid back into YouTube kids: detecting inappropriate content on video streaming platforms. In: Proceedings of the IEEE/ACM international conference on advances in social networks analysis and mining, (pp 464–469)
34. Papadamou K, Papasavva A, Zannettou S, Blackburn J, Kourtellis N, Leontiadis I, Stringhini G, Sirivianos M (2020) Disturbed YouTube for kids: characterizing and detecting inappropriate videos targeting young children. In: Proceedings of the international AAAI conference on web and social media (vol 14, pp 522–533)
35. Neumann MM, Herodotou C (2020) Young children and YouTube: a global phenomenon. Child Educ 96(4):72–77
36. Ghosh AK, Badillo-Urquiola K, Guha S, LaViolaJr JJ, Wisniewski PJ (2018) Safety vs. surveillance: what children have to say about mobile apps for parental control. In: Proceedings of the CHI conference on human factors in computing systems, (pp 1–14)
37. Yadav S, Chakraborty P (2021) Child–smartphone interaction: relevance and positive and negative implications. Universal access in the information society, in press
38. Yadav S, Chakraborty P (2021) Children and new media: can playing with smartphones be beneficial? Media Asia, in press

Cold start and Data Sparsity Problems in Recommender System: A Concise Review

M. Nanthini and K. Pradeep Mohan Kumar

Abstract An enormous amount of data available on the e-commerce sites are of different forms as ratings, reviews, opinions, remarks, feedback, and comments about any item, and it is difficult for the system to search the user interest and predict the user preference. The recommender system (RS) came into existence and supports both customers and providers in their decision-making process. Nowadays, recommender systems are suffering from various problems such as data sparsity, cold start, scalability, synonymy, gray sheep, and data imbalance. One of the major problems to be considered for better recommendation is data sparsity. Cross-domain recommendation (CDR) is one way to address data sparsity problems, cold start issues, etc. In the most traditional system, cross-domain analysis is used to understand the feedback matrices by transferring hidden information and imposing dependencies across the domains. There is no vast comparison of existing research in CDR. This paper defines the problem, related and existing work on CDR for data sparsity and cold start, comparative survey to classify and analyze the revised work.

Keywords Cross-domain recommendation · Collaborative filtering · Recommender system · Data sparsity · Cold start

1 Introduction

RS is an information filtering system that tries to find the rating or preference given by the customer to the product on e-commerce Web sites. The most traditional recommendation methodologies are collaborative filtering (CF) and content-based filtering. Collaborative recommendation [1] is also known as a social recommendation, and it provides recommendations based on similar people's preferences, likes, dislikes, and

M. Nanthini (✉) · K. Pradeep Mohan Kumar
Department of Computing Technologies, SRM Institute of Science and Technology, Kattankulathur, Chennai, Tamil Nadu, India
e-mail: nm7342@srmist.edu.in

K. Pradeep Mohan Kumar
e-mail: pradeepk@srmist.edu.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_9

reviews. The content-based recommendation is cognitive filtering, and it provides recommendations according to the content of an item and the user profile. Hybrid recommendation systems combine the popular two approaches collaborative and content-based systems by collecting the user profiles learning the model based on the information and maintaining some useful records using efficient information retrieval techniques and other content-based methods, and directly comparing the user profiles to find similar users based on collaborative filtering [2] and provide recommendations. The above mentioned filtering represents that items can be suggested to the users in the form of a recommendation list when items' score is higher than the user's profile or are users' higher rating with a similar profile.

Monolithic hybridization is a technique that represents various levels of recommendation techniques in one algorithm while implementation. The single recommender component combines multiple techniques by pre-analyzing, processing and integrating various information sources. Hybridization is attained by making algorithm changes to derive various kinds of input data. Figure 1 represents the types of RS in various perspectives. In traditional systems, filtering techniques are providing a recommendation based on single-domain information. The domain is one of the characteristics of a recommender system, and it is a particular thought of field, activity, or user interest.

Considering two domains in RS, the numbers of attributes to be considered are larger and different compared to the single domain. Dimensionality reduction (DR) [3] helps to reduce the overall dimensionality in two domains and increase the prediction accuracy of user preference in RS by discovering the hidden knowledge of the whole data set and transferring knowledge across the domains. The data available

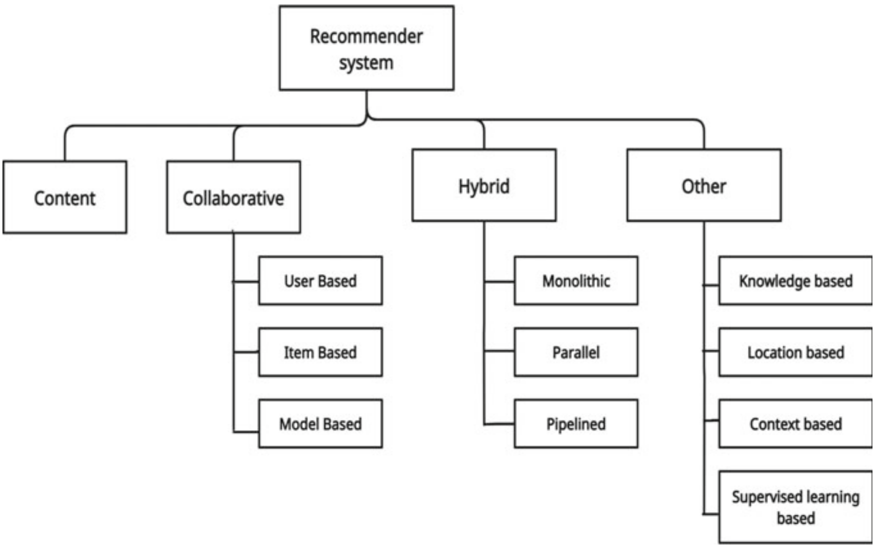


Fig.1 Types of recommender system

on the online Web sites are of various types such as continuous/discrete, numerical/categorical, and structured/unstructured. Continuous data are the range that can take any value between its maximum and minimum value. Structured data refers to the information in a fully organized manner and can enable a quick search algorithm. Unstructured data have a lack of structure which makes it an energy-consuming task. Categorical data can take only certain values and is also known as discrete data. It can be classified as nominal or ordinal variables. One of the challenges is analyzing such a variety of data, and DR techniques can be applied for data analytics.

2 Related Work

This section discussed various related research on CDR and sparsity problems in other RS and explained the ways to address the problem faced by traditional RS.

2.1 Data Sparsity and Cold start in CDR

In an online social network, the star-structured hybrid graph is used for making recommendations across the domains. The useful information is shared from enriched domain to domain is having sparse data. Different factors such as the transfer of item feature considering popularity and consistency of the behavior are explored. Jiang et al. introduced a method called hybrid random walk (HRW) [4], which considers the above factors to identify the items that can be transferable and share the knowledge between two different domains and also identify the relationships between the user and item. A CF-based cross-domain algorithm is developed to build a linear decomposition model (LDM) by combining items to find relationships among total and local similarities of multiple domains [5]. Yu et al. proposed an algorithm to compute the whole similarities in all considered domains, and it is the fusion of various domains' local similarities. The weights are computed using two domains and greater than the weight in a single domain, and it is reflected in the similarity measures in the source domain. If the user preferences vary, it is difficult to do better recommendations with multiple domains.

A CDR is constructed based on attributes and improper classification. The traditional recommendation issue is considered as a rough unevenness categorization in the target domain which took feature vector of item and user and rating as a label. Then, the sparsity problem is reduced by considering the useful vector of item and user in the target domain. Yu et al. here they have used singular Value Decomposition (SVD) to extract the useful features about users from two source domains for a better recommendation. Later, an unevenness classification model [6] is constructed to rectify an unevenness classification problem that can efficiently solve the unequal distribution of ratings. Data sparsity is one of the important problems in RS, and it is dominant in newly constructed RS which is having insufficient data. CDR is considered an efficient solution to the sparse data problem by transferring sufficient information across the domains. While transferring information from one domain to another domain, three different cases were explored such as entities that are fully overlapped, partially overlapped, and non-overlapped. Even though the entities are fully overlapped, they are having different formats in each domain. The above cases reduced the overall performance of CDR in the target domain. Zhang et al. proposed a Kernel-induced [7] RS based on knowledge transfer which interrelates the entities that are non-overlapped in two domains to solve the sparse data problem in RS.

Transferring knowledge from data enrich domain to domain is having sparse data play a vital role in addressing the data sparsity problem in RS. Source domain is enriched with sufficient data, whereas the target domain used the information from the source domain for a good recommendation. Model-based on cluster-level rating is used in this work to investigate the sparsity problem without considering the overlapping entities in two domains. The existing methods have not addressed the accuracy of expected results in the target domain. Ming He et al. proposed a model called an adaptive codebook transfer learning (ACTL) [8] to develop the codebook scale-based technique which reduces the cost for computation and improves the prediction accuracy and features size and attributes of the source domain in CDR.

The alternate course selection process in colleges and universities is a difficult task for the students due to its unknown nature among the students. This leads to inappropriate alternate course selection and low achievements in the course which compels the students to quit the course halfway. To address the unknown nature of course and improve their selection process, Huang et al. a cross-user domain CF [9] is developed for the exact prediction of marks scored by each student with the help of score distribution scored by senior students. Then, the top alternate courses which are having high marks scored by the senior students are recommended to the present students for their achievement. A citation-based recommendation in the cross-domain analysis [10] is developed for knowledge sharing. A cross-domain recommendation has evolved to help users in detecting similar knowledge across the domains and provide recommendations based on the shared information. The first technique is simple keyword mapping. The second technique is co-citation selection.

In addition to data sparsity, data imbalance is considered a challenging problem in CDR while transferring knowledge across the domains. This research work solves the above problem using three different learning methods such as representation, adversarial, and transfer. Even though different transfer learning techniques performed well in this area, a novel RecSys-discriminative adversarial network (DAN) concentrated on multiple problems such as data sparsity inside the domain and across the domain, data imbalance, and knowledge transfer in the form of latent factors in CDR. The knowledge transfer is performed in an adversarial manner [11]. Four different neural architectures are developed and explored. Real-time experimentation showed that the proposed technique performed well in the target domain without labeled data, and it is more flexible in multiple real-world scenarios and robust to cold start problems in RS.

Nowadays, online commercial businesses are more dependent on RS to identify the individual user's needs and provide them with better recommendations. To provide efficient recommendations, a large amount of data with knowledge about the user and items are needed. By sharing an enormous amount of data from one domain to another domain, the users–items relationship was explored to complement the CF recommendation with greater prediction accuracy. However, the efficient utilization of information across the domains is a difficult problem. Do et al. proposed to explore latent features with similarities in multiple domains based on matrix factorization [12]. In this research work, features that are common in two domains and the features which are specific to the domains are analyzed. Both the information are very useful in CDR. Especially, the domain-specific features are considered to transfer the knowledge across the domain.

CDR is an efficient system to address the sparsity problem in RS by considering knowledge from multiple domains. Hong et al. proposed a deep neural network based on the cross-domain technique [13] for a good recommendation. The proposed technique rectifies the prediction problem based on rating with the help of metadata information including reviews and items. The learning process happened in the target domain as well as in source domains with hidden factors of users and products which increase the prediction accuracy. The mapping process and the optimization experimented in both the domains solve the sparsity problem effectively in the target domain. CF and matrix factorization are failed to perform well in RS having no information about the user and the product that reduce the income of the online commercial business. It is a very challenging task to provide a recommendation to the new user. Jin et al. proposed a novel review aware recommendation [14] using cross-domain analysis to enquire about the new user recommendation in the E-commerce sites. In this work, reviews are collected using an adjacency matrix and the preference vectors are taken out from the domains in terms of specific and shared manner using migration model.

The cold start problem is also an important issue in RS having new users and items. RS with cold start users reduces the entire performance of the system. To address the above problem, Wang et al. proposed a technique that merges the e-shopping domain and the information from Ads. A CDR is developed using the deep learning algorithm; word to vector concept is employed to convert text information into latent information. Deep learning algorithms strengthen the performance of RS for sharing the knowledge from source to target domain. Text data are having an enormous amount of knowledge compared to latent representation; R-metapath2Vec [15] is used to enhance the features in latent space. The existing studies on CDR are mainly focused on knowledge transfer from one domain to another domain on the same Web site. It is very difficult to share full information across the domain in different e-commerce sites due to some privacy concerns, and it leads to negative transfer problems. Zhang et al. proposed an RS named selective knowledge transfer [16] that shares both hidden features of users and products from data enrich domain to sparse data domain and also addresses the negative transfer problem.

2.2 Collaborative Filtering Recommendation

The social recommender system [17] uses collaborative filtering (CF) to make personalized recommendations. Most of the CF techniques mainly use the one-dimensional clustering methods to group users and items separately. However, the one-dimensional method usually neglects the necessary information in another dimension. Zhang et al. used the bi-clustering technique that groups both the user features and item features concurrently in the user–item matrix. Correspondingly, the bi-clustering method outperforms the one-way cluster method of overcoming the sparsity problem and reducing the high-dimensional matrices in recommender systems. The recommendation in social networks gains more popularity and attaining a successful service for providing quality of service and user satisfaction. These applications enable the users to provide various implicit feedbacks and ratings on the web during their daily usage of the social network. The users interacted in social networks like some items based on traditional RS filtering techniques and can provide feedback about the item. With the exponential growth of online information, the sparse data gradually decrease the overall performance of RS quality and correlation factor of feedbacks. A novel location-aware RS [18] is developed, and it is a similarity measurement-based approach to calculate the similarity between the paths and recommends products based on spatial–temporal pattern discovery that makes the shopping environment more intelligent.

3 Overview of CDR and Domain Level

This section describes about cross-domain analysis, latent variables. Attributes mapping from user profiles to the latent variable. And also detail domain-level attributes mapping to the latent variables.

3.1 Cross-domain Analysis

A domain is considered as an action or user preference. Most of the RS focused on one domain, and they suggest recommendations within the domain where users gave their feedback and ranks. The combination of various domains into a single domain makes the recommender engine provide a recommendation list across the domain. An algorithm used in domain analysis can provide a recommendation of the product in the chosen domain to users who gave ratings only in the source domain. For example, if the restaurant is referred to as the source domain and the tourist spot is referred to as the chosen domain, the knowledge is shared across the restaurant domain and tourist spot domain.

3.2 Various Domain Level

- Attribute level (Chinese cuisine ↔ Asian cuisine)
- Cuisines share the same kind of attributes with different values for some attributes.
- Type-level (restaurants ↔ tourist spot)
- Different types, sharing some common features like location, climate.
- Item level (shopping ↔ restaurants)
- An entirely different set of features and attributes.
- System level (TripAdvisor ↔ MakeMyTrip)
- This level is having the same kind of items and is collected and expressed in different ways and manners, respectively.

3.3 Different Approaches in CDR

Knowledge aggregation is the process of combining user interests, likes, and preferences from across the domains and providing recommendations in the target domain. User preferences can be clicks, ratings, logs, and feedbacks which are merged for performing knowledge linking. It is very useful to address the cold start problem.

Figure 2 represents a knowledge aggregation process using two domains. Knowledge transfer is the process of sharing knowledge using latent features and ratings across domains. Figure 3 shows that the common latent factors are used for knowledge transfer.

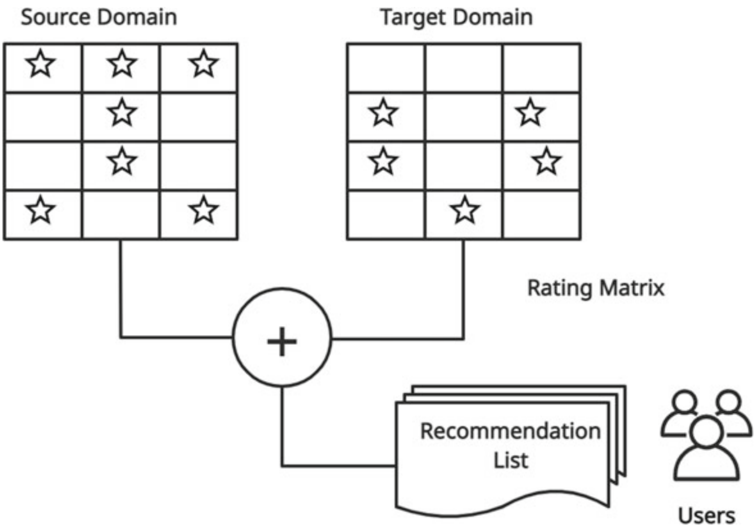


Fig. 2 Linking knowledge

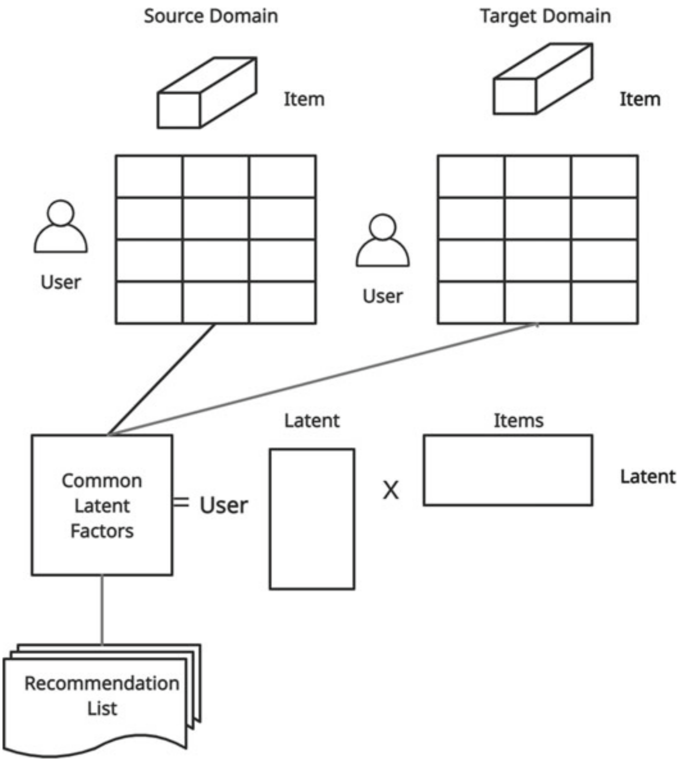


Fig. 3 Knowledge transfer

3.4 Latent Variables

Latent variables are the variables that are not present in the dataset but are inferred from the variables that are present in the original dataset (observed variables). By introducing the set of latent variables, unobserved inferences from the real-world datasets are discovered and uncover the hidden patterns that lead to knowledge discovery. Mathematical models that aim to derive the latent variables from observed variables are called latent variable models. One advantage of using a latent variable is a dimensionality reduction in big data. An enormous amount of collected variables can be merged in a model to reproduce a hidden concept, making it easier for better understanding of data. The dimensionality reduction is the process of decreasing the number of collected variables and can be classified into feature selection and feature extraction.

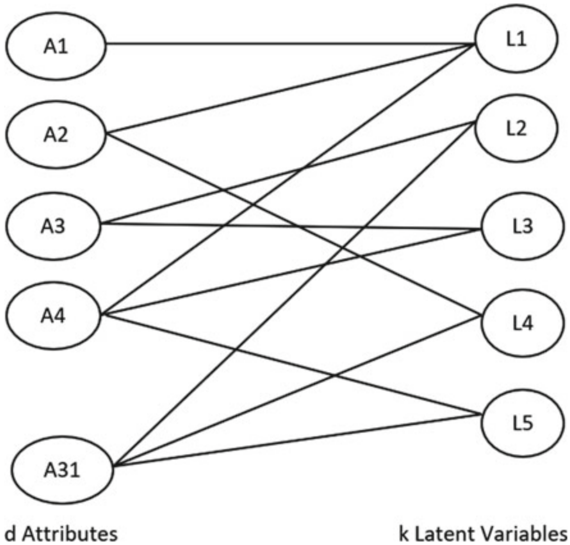


Fig. 4 Dimensionality reduction

3.5 Attributes Mapping to Latent Variables

The system uses two domain-level information and user profiles for attribute mapping. The two domains considered for dimensionality reduction in RS are restaurants and tourist spot [19]. Common features across the domain can be extracted using latent class analysis. Many-to-one mapping of attributes to latent variables is done to reduce the dimensionality of overall attributes in a dataset considering related domains. The user profiles are name, location, gender, history, badges collection, places visited, travel styles, contribution, price range, date of visit, ratings, and reviews. Restaurant attributes are name, location, restaurant ratings, reviews by the user, cuisine, and review count. Tourist spot attributes are name, ratings, tourist spot reviews, spot location, and the type. Figure 4 represents the dimensionality reduction of "*d*" attributes to "*k*" latent variables. The latent factors are classified into the corresponding label for the learning process. The general equation for latent variable derivation from observed attributes is given as Eq. (1)

$$\text{Lat} = \cup(\sum(f(\text{user}),\ f(\text{rest}), f(\text{tour}))) \tag{1}$$

where f(user), f(rest), and f(tour) are the functions for user profiles, restaurants, and tourist spots, respectively, which are mapped from various observed variables corresponding to the latent variables.

Table 1 Summary of related work

Comparative summary on cross-domain recommendation				
Study	Major attention	Cold start addressed	Data sparsity addressed	Open access
[4]	HRW with the user–item link in the target domain	Yes	Yes	No
[5]	User-based CDR with LDM	No	Yes	Yes
[6]	Funk SVD-based CDR for multimedia application	No	Yes	Yes
[7]	Kernel induced RS for overlapping entities	No	Yes	No
[8]	ACTL for CDR	No	Yes	Yes
[9]	Score prediction-based course recommendation using cross-user domain CF	No	No	Yes
[10]	Co-citation selection-based CDR	No	No	No
[11]	RecSys-DAN-based CDR	Yes	Yes	No
[12]	Matrix tri-factorization-based CDR	No	No	No
[13]	Deep neural network-based CDR	No	Yes	Yes
[14]	Review-aware CDR	Yes	No	Yes
[15]	DNN-based CDR	Yes	No	Yes
[16]	Selective knowledge transfer for CDR	No	Yes	Yes
[19]	Latent variable-based recommender system	No	Yes	Yes

4 Comparative Discussion

Table 1 represents the comprehensive survey of various techniques used in CDR for addressing sparsity of data and new user problems in CDR, open access of the journal, and its scope. Figure 5 represents the survey charts of data sparsity and cold start issues addressed.

5 Conclusion

The most prevalent problems in RS are sparse data and new user issues. The social networks allow the users to construct relationships among the various types of items across the domains and provide a recommendation list to the users and to address

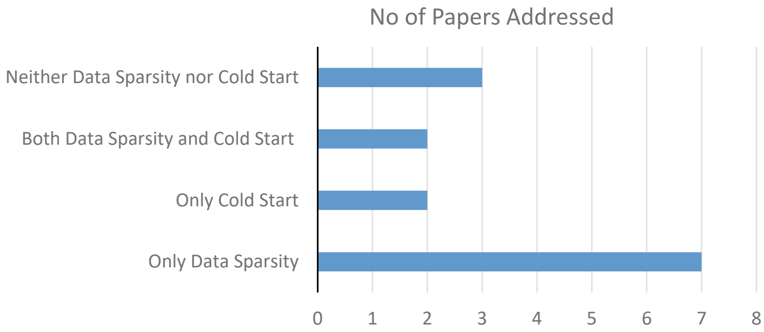


Fig. 5 Survey chart of cold start and data sparsity addressed

the sparse data and new user problems. The CDR is explored to address the above problems with different types of techniques from various perspectives. This work also discusses the knowledge transfer from auxiliary domain to final domain, knowledge linking with multiple domains, latent variable analysis for cross-domain technique, and attribute mapping into low-dimensional data for a better recommendation. Most of the CDR is based on traditional techniques, and it becomes less efficient when the system receives dynamic data in a real-time environment. In the future direction, deep learning-based CDR will be explored to achieve high performance compared to the traditional techniques.

References

1. Xu Y, Peng Q, Lingwei X, Jiang F, Junwei D, Gong D (2021) A selective ensemble learning based feature two-sided cross-domain collaborative filtering algorithm. Inf Process Manage 58(6):1–12
2. Cui Z, Xianghua X, Xue F, Cai X, Cao Y, Zhang W, Chen J (2020) Personalized recommendation system based on collaborative filtering for IoT scenarios. IEEE Trans Serv Comput 13:685–695
3. Zebari R, Abdulazeez A, Zeebaree D, Zebari D, Saeed JA (2020) A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction. J Appl Sci Technol Trends 1(2):56–70
4. Jiang M, Cui P, Chen X, Wang F (2015) Social recommendation with cross-domain transferable knowledge. IEEE Trans Knowl Data Eng 27(11):1041–4347
5. Xu Y, Jiang F, Junwei D, Gong D (2017) A user-based cross domain collaborative filtering algorithm based on a linear decomposition model. IEEE Access 5:27582–27589
6. Yu X, Fu Y, Xu L, Liu G (2018) A cross-domain recommendation algorithm for D2D multimedia application systems. IEEE Access 6:62574–62583
7. Zhang Q, Lu J, Wu D, Zhang G (2019) A cross-domain recommender system with kernel-induced knowledge transfer for overlapping entities. IEEE Trans Neural Networks Learn Syst 30(7)
8. He M, Zhang J, Zhang S (2019) ACTL: adaptive codebook transfer learning for cross-domain recommendation. IEEE Access 7:19539–19549

9. Huang L, Wang CD, Chao HY, Lai JH, Philip SY (2019) A score prediction approach for optional course recommendation via cross-user-domain collaborative filtering. IEEE Access 7:19550–19563
10. Tantanasiriwong S, Haruechaiyasak C (2014) Cross domain citation recommendation based on co-citation selection. IEEE Trans Cybern 8(6):978–987
11. Wang C, Niepert M, Li H (2020) RecSys-DAN: discriminative adversarial networks for cross-domain recommender systems. IEEE Trans Neural Networks Learn Systems 31(8):2731–2740
12. Do Q, Liu W, Fan J, Tao D (2021) Unveiling hidden implicit similarities for cross-domain recommendation. IEEE Trans Knowl Data Eng 33(1):302–315
13. Hong W, Zheng N, Xiong Z, Zhiqiang H (2020) A parallel deep neural network using reviews and item metadata for cross-domain recommendation. IEEE Access 8:41774–41783
14. Jin Y, Dong S, Cai Y, Jinlong H (2020) RACRec: review aware cross-domain recommendation for fully-cold-start user. IEEE Access 8:55032–55504
15. Wang H, Amagata D, Makeawa T, Hara T, Hao N, Yonekawa K, Kurokawa M (2020) A DNN-based cross-domain recommender system for alleviating cold-start problem in E-commerce. IEEE Open J Ind Electron Soc 1:194–206
16. Zhang H, Kong X, Zhang Y (2021) Selective knowledge transfer for cross-domain collaborative recommendation. IEEE Access 9:48039–48051
17. Zhang D, Hsu CH, Chen M, Chen Q, Xiong N, Lloret J (2014) Cold-start recommendation using bi-clustering and fusion for large-scale social recommender systems. IEEE Trans Emerg Top Comput 2(2):239–250
18. Zeng D, Liu Y, Yan P, Yang Y (2021) Location-aware real-time recommender systems for brick-and-mortar retailers. Informs J Comput 33(4):1608–1623
19. Valliyammai C, Nanthini M, Thendral SE (2016) Dimensionality reduction using latent variable across the domains in recommender system. Int Res J Electron Comput Eng 2(2):33–37

A Hybrid Approach to Find COVID-19 Related Lung Infection Utilizing 2-Bit Image Processing

Md. Ashiq Mahmood, Tamal Joyti Roy, Md. Ashiqul Amin, Diti Roy, Aninda Mohanta, Fatama Fayez Dipty, and Shovon Mitra

Abstract This study describes the deployment of an image processing approach for finding COVID-19 affected lungs. Medical scans are useful in diagnosing illnesses and determining if organs are working normally. Medical image processing is an ongoing research subject in where numerous ways are used to help diagnosis, as well as different image processing techniques that may be used. Picture processing was used in this work, which includes image pretreatment, histogram leveling, smothering, eroding, and dilation. The usage of 2-bit picture is selected since this characteristic is well-known and there are several resources accessible. The Open CV library, which includes a plethora of image processing functions, is likewise free to use. Our experiment has shown how COVID-19 affected lung disorders can easily be identified with the help of a 2-bit image segmentation technique. The plan comprises (1) using a deep robust acquisition access to portion proper regions of interest from bleak medical examination image sizes of 903 total, (2) using a propagative neural network to improve contrast, sharpness, and illuminance of image contents, and (3) from the beginning to the conclusion, a regression strategy plan was used to accomplish medical picture categorization by material design in deep neural networks.

Keywords Image processing · Bio-informatics · Neural network · Artificial intelligence · COVID-19

Md. Ashiq Mahmood (✉) · T. Joyti Roy · D. Roy
Institute of Information and Communication Technology, Khulna University of Engineering and Technology, Khulna, Bangladesh
e-mail: ashiqmahmoodbipu@gmail.com

Md. Ashiqul Amin
Northern Arizona University, Flagstaff, AZ, USA
e-mail: ma3359@nau.edu

A. Mohanta
Institute of Engineering and Management, Kolkata, India

F. Fayez Dipty
North Western University, Khulna, Bangladesh

S. Mitra
Brac University, Dhaka, Bangladesh

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_10

1 Introduction

Nowadays, the coronavirus (COVID-19) has outspread out all over the Earth. The citizenry of all world are suffering from this pandemic. They are affected by fear, anxiety, depression, and loneliness. Not only fear but also economic crisis is a big challenge for them [1]. Most people are suffering from the disease because of the COVID-19 pandemic, and sometimes, it is very difficult to predict the symptoms of some diseases. But, the symptoms can be detected by using the image processing method that is vastly used in the modern era. To get much information from an image, this process is used, and it is a method to convert an image into digital form. Image processing can be used to predict many diseases like crop disease, plant disease, leap disease, skin disease, fungal disease, lung cancer, etc. The lungs are the organs that are mostly affected by the coronavirus(Sars-Cov-2). The tubule or cartilaginous tube is depicted by the snout. Into your lungs, it separates into little and small branches. Alveolar are tiny air sacs that may be seen at the tips of each branch. That is where O_2 enters and CO_2 departs your blood. Sometimes, it is difficult to detect but image processing can be used to detect these symptoms in the initial stage, and it is a very helpful and low-cost system. Your immune system fights back as the virus spreads direct your metabolic process. The lungs and airways expand and inflame. This might begin in one area of your lung and progress to the rest. COVID-19 causes are mild to severe symptoms in around 80% of individuals. You might be suffering from a dry cough or a sore throat. Pneumonia, a lung illness in which the alveoli become inflamed, affects certain people. As lungs are mostly affected by COVID-19, so our paper aims to predict lung disease which is influenced by this coronavirus. We have used 900 images. Our image dataset consisted of patients who were affected with COVID-19 and other lung diseases during the 2020–2021 time region.

2 Literature Review

The goal of this study [2] is to provide a conciliate informing to large learning method in medical image model processing, starting with abstract underpinnings and working through applications. In this [3] experiment, they explored cutting-edge deep learning architecture and how to optimize it for medical picture segmentation and classification. This article explained how to use the Python3 and via the Open CV package to process medical pictures[4]. Many techniques had been discussed in this literature [5], among those there were ROI segmentation, K-means, and water-shed. The primary goal of this [6] the goal of this research was to create a comprehensive source for healthcare visual analytic methods that have benefited from high-performance computer platforms. With this in mind, publications from the Scopus and Web of Science electronic archives were searched. Sobel and Prewitt used algorithms in their project. [7] edge detection research using 180 nm technology. VLSI is used to construct the edge detection algorithms, and the architecture's digital IC design is discussed. The authors of this study suggested a flexible framework [8] for deep eruptive acquisition-based medical image processing and analysis. By incorporating [9] motorist memorability into PCNN, the network gains biological function. Furthermore, the use of nanosecond resistors could considerably reduce the size of PCNN.

3 Methodology

Image processing has had lots of algorithms and processing tools. Our experiment established the process of judgment based on image analysis. We have conducted our experiment by taking data from this source [10]. Our data was divided into four categories, for example, COVID-19 infected patients, bacterial infected patients, age-related lung failure patients, and some of them were idiopathic [11]. Figure 1 shows our whole experiment in a nutshell. We have used the 903 images where 70% of images were COVID-19 affected. 10% were bacterial infections, 5% were age-related lung failure, and 15% were idiopathic. We have used the Python programming language for our experimental analysis. First, we have analyzed without converting the images into segments of binary values. The findings of RGB pictures

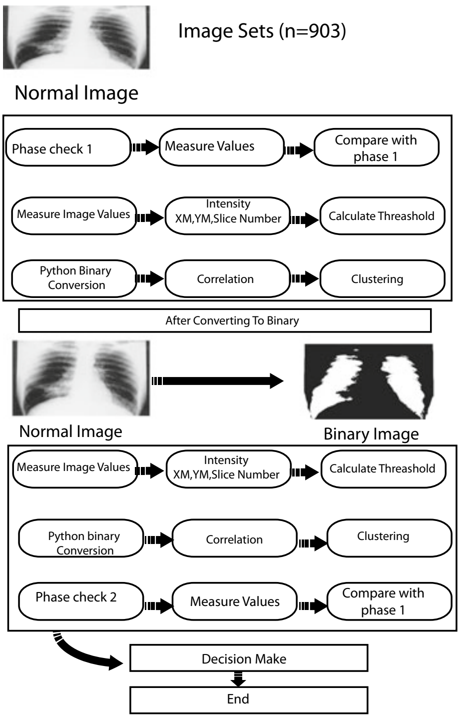


Fig. 1 Experimental flowchart showing from the initial data collection where the total number of images was 903, checked phase one and measure the values, compare again with initial stages then again after measuring the XM, YM and slicing started, the below portion showing after the converting the normal image into binary. Lastly, the whole data was again checked, and decisions have been made and stop the process

are computed using brightness values. The formula is used to convert RGB pixels to brightness values and calculates and shows the mean, standard deviation, minimum, and maximum values for each column. Secondly, we have converted our normal images to binary and then again gathered the values. This time the values showed significant changes. The %area denoted that how much the Human Lung is affected. The more the area cover, the more infected the individual's lung is. We also used parameter-based neural network systems for validating our second result to make sure all our experiments were okay. Nevertheless, the correlation showed that there was enough evidence to show the COVID-19 patient's lung was much more vulnerable and affected than the normal age-related infected lung or lung infection caused by bacteria. The second iteration took more than three minutes in the programming environment, in this case, Jupiter notebook. The model training sum of squares error was 15.821, and average overall relative error was 0.428. The testing period consisted of 8.191 sums of squares error and 0.469 average overall relative error. The area percentage fluctuated from −0.453 to 1.375. All the 903 images were considered. The last iteration showed fluctuation between 0.002 and 0.394 (Tables 1 and 2).

Table 1 Sample of data set view

Label	*X*	*Y*	XM	YM	%Area
COVID-19	469.795	241.408	469.795	241.408	27.741
COVID-19	523.168	543.949	523.168	543.949	52.743
COVID-19	460.219	336.42	460.219	336.42	44.572
Bacterial infection	1072.626	916.776	1072.626	916.776	41.874
COVID-19	421.712	571.308	421.712	571.308	40.221
COVID-19	195.925	183.731	195.925	183.731	43.638
Bacterial infection	401.952	364.924	401.952	364.924	51.856
COVID-19	1267.149	1712.241	1267.149	1712.241	43.404
Bacterial infection	569.687	532.223	569.687	532.223	49.648
COVID-19	2317.007	2214.071	2317.007	2214.071	53.804

Table 2 Layer optimization values

Layers		H(1:1)	H(1:2)	H(1:3)	H(1:4)	Area	Area_A
Input layer	(Bias)	1.036	0.388	0.708	−0.296		
	X	−0.164	−0.476	−0.018	−0.323		
	Y	0.013	1.042	−0.023	−0.082		
	XM	−0.460	−0.204	−0.677	0.389		
	YM	−0.089	0.650	0.320	−0.376		
Hidden layer 1	(Bias)					−0.453	1.375
	H(1:1)					0.044	−1.352

Table 3 Model description

Training set	Sum of squares error	15.821
	Average overall relative error	0.428
	Relative error for scale dependents	0.763
Testing set	Training time duration	0:00:00.02
	Sum of squares error	8.191
	Average overall relative error	0.469
	Relative error for scale dependents	0.681

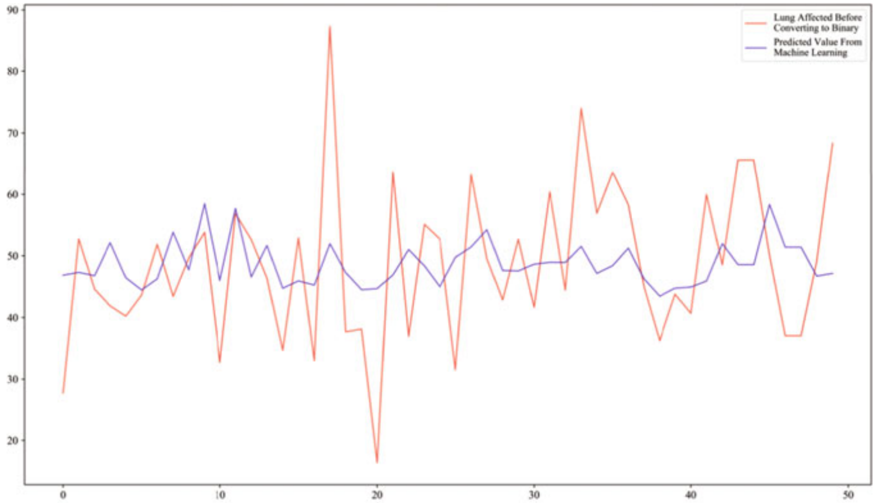
3.1 Image Processing Terms Description

Threashold is the image's threshold. To get a binary picture *B* with 1 in blank pixels and 0 in character dots, apply a 0.9 threshold to *I*. To conduct binary erosion on *I*, we utilized a 2 10 horizontal mask. This helps to propagate the character pixels along the rows, resulting in a significant disparity between the sums corresponding to blank rows and rows containing characters when we 1 add up the rows of the photo. To generate a list of m numbers, add the horizontally degraded image along rows. Many blank rows appear adjacent to one another, as one might expect. The image is broken into multiple images, each carrying one row of text, at the mid-point of each continuous stretch. The midpoints of contiguous stretches of blank columns are $m1$, $m2$, ..., mp. For $I = 1$ to $p = 1$, the pictures $I[mi\colon mi + 1, :]$ contain a row of text. To get column splits and further separate each text row image, we conducted steps 2–4 for the picture of each line using a vertical mask instead of a horizontal mask and column sums instead of row sums. Most of the characters are split; however, if the text contains characters like I or subscripts in, we may not have divided each character. As a result, if the image acquired after step 5 has more than one linked component, steps 2–5 are repeated (Table 3).

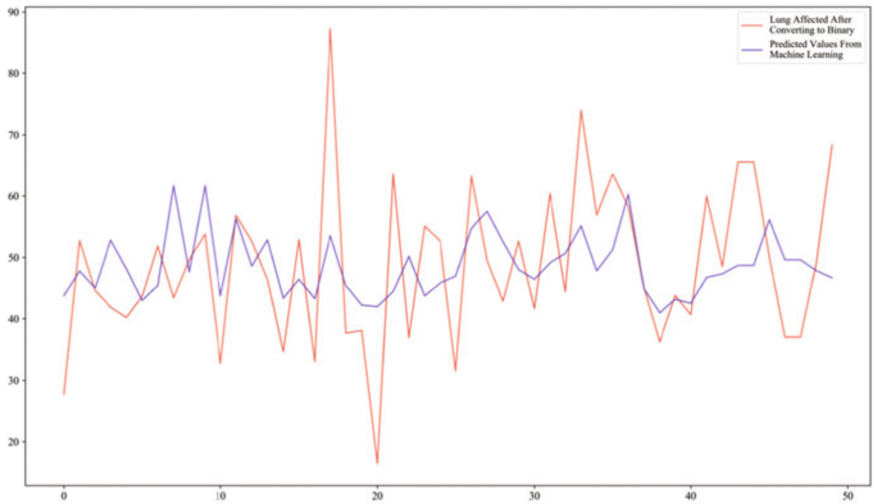
$$\text{ValueImage} = 0.299\,\text{Red} + 0.587\,\text{Green} + 0.114\,\text{Blue} \tag{1}$$

4 Experimental Results

Figures 2 and 3 showing the initial phase of conducting experiments in binary environments. The after results clearly showing the more enlarged area. The normal image could not give as many of the binary values. We were able to create a theoretically novel and possibly helpful metric for comparing 2-bit secondary structures by drawing inspiration from image processing and the dot plot representation for 2-bit secondary structure. We demonstrated our method using the 2-bit design issue and an application that uses the distance measure to discover informational rearrangement point mutations in a picture sequence. The process of restoring a picture from a



(a)



(b)

Fig. 2 **a** Before converting to binary, the lung affected number was constant. The machine learning prediction curve shows the constant rate. **b** After converting to binary, this time the affected lung images did not show any constant curve rather showed ups and downs that means there were infected lungs and detectable after processing

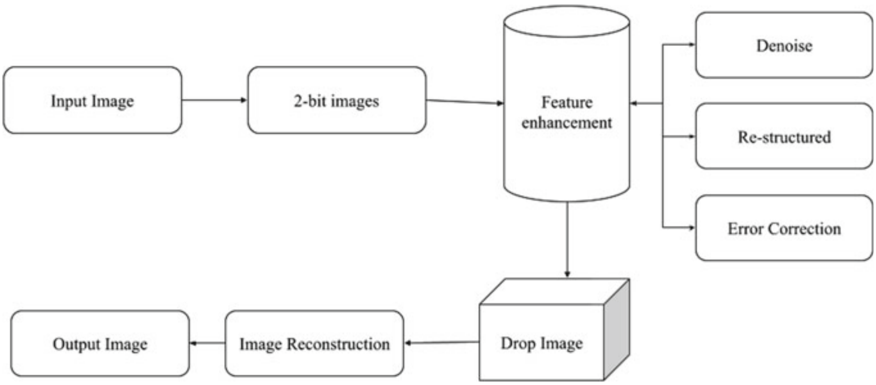
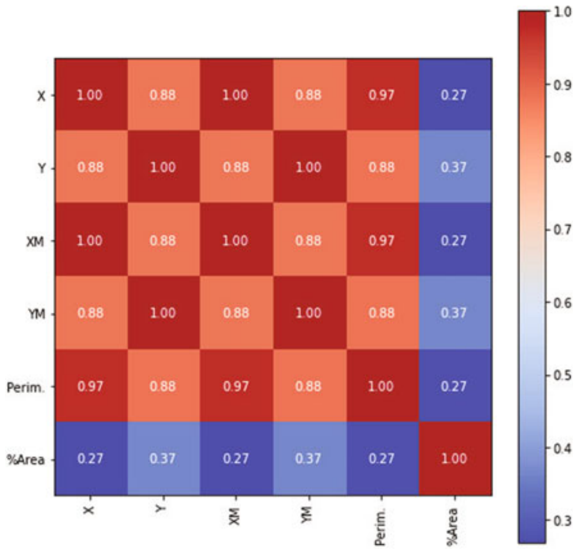


Fig. 3 Image restoration, 2-bit images once again collected after processing and features were again extracted in reverse form thus whole images can be seen. Denoise algorithm applied and again compare with the real images

Fig. 4 Correlation matrix showing the XM, YM, area and *X*, *Y* value intensity



damaged version-typically image restoration is the process of restoring a blurry and noisy image. Image restoration is a fundamental topic in image processing that may also be used to evaluate more general inverse problems. The restored image's quality, the method's computing efficiency, and the estimation of critical parameters like the point-spread function are all factors to consider are all critical concerns that must be addressed, Fig. 4 showing our image restoration procedure. To offer an understanding of the nature of the problem, basic picture restoration techniques are reviewed. These approaches also give efficient solutions for deblurring hazy pictures with a low computing complexity. A similarity matrix is a tool that displays the collinearity

Table 4 Acronym used

Acronym	Meaning
ROI	Region of interest
VLSI	Very large scale integration
PTL	Paced transfer learning
PCNN	Partially connected neural network
ITK	Indigenous traditional knowledge
VTK	Virtual tool kit
Binary	Values consisted of 0 and 1
OpenCV	OpenCV is a programming function library primarily
CATIA	Dassault Systèmes' multi-platform software package
X	Value of X co-ordinate
Y	Value of Y co-ordinate
XM	Intensity of X co-ordinate pixel
YM	Intensity of Y co-ordinate pixel

for several variables. The matrix illustrated in Fig. 5 shows the correlation among the different attributes like x, XM, y, YM, intensity, and percent area. It is a strong tool for summarizing a immense datasets as well as distinguishing and visualizing trends in the data (Table 4).

Algorithm 1 Input

1: Start
2: initial labeled images, composed of 903 samples
3: calculate the x,xm,y,ym
4: train the segmentation of phase-1
5: test the segmentation of phase-2
6: make image centroid_initial with $k_i = R_{i+1}$
7: for reoccurance_i:
8: initialize $k_i + 1 = R_{i+1} + 1$
9: End

5 Conclusion

Images are a figurative way of expressing data. Images are made up of tiny components known as pixels. Each pixel has a incomparable function and treasure, and we wanted to deal with these values. In our example, it is a two-bit binary picture. A geometric picture refers to an image that is represented arithmetically by non-representational schema such as lines. Each photograph is saved in a specific file format, which consisted of two parts: the header and the data. Imaging processing methods are a set of ways for manipulating images with the help of a computer. We were able to implement a deep robust learning approach for our analysis using a prerogative neural network that helps to improve the contrast, sharpness, and illuminance of the images. We also made it possible by using the regression strategic plan to perform scanned lungs images to classify. The goal of segmentation is to divide pictures into significant sections. The partitioning of pictures is handled via local segmentation. The main limitation of our analysis is that if we could analyze for more than a million data images, then there would be more chances of prediction of COVID-19 affected lung. In the future, 4-bit or 8-bit image analysis could be made possible.

References

1. Roy D, Roy TJ, Mahmood MA (221) An efficient approach to identify economic crisis during covid-19 outbreaks utilizing data mining. SSRN Electron J. https://doi.org/10.2139/SSRN.3852813
2. Maier A, Syben C, Lasser T, Riess C (2019) A gentle introduction to deep learning in medical image processing. Z Med Phys 29(2):86–101. https://doi.org/10.1016/J.ZEMEDI.2018.12.003
3. Razzak MI, Naz S, Zaib A (2018) Deep learning for medical image processing: overview, challenges and the future. Lect Notes Comput Vis Biomech 26:323–350. https://doi.org/10.1007/978-3-319-65981-7_12
4. Widodo CE, Adi K, Gernowo R (2020) Medical image processing using python and OpenCV. J Phys Conf Ser 1524(1):012003. https://doi.org/10.1088/1742-6596/1524/1/012003
5. Mohamed Y, Abdallah Y, Alqahtani T (2019) Research in medical imaging using image processing techniques. In: Medical imaging—Principles and application [working title], Jun 2019. https://doi.org/10.5772/INTECHOPEN.84360
6. Gulo CASJ, Sementille AC, Tavares JMRS (2017) Techniques of medical image processing and analysis accelerated by high-performance computing: a systematic literature review. J Real-Time Image Process 166, 16(6):1891–1908. https://doi.org/10.1007/S11554-017-0734-Z
7. Vardhana M, Arunkumar N, Lasrado S, Abdulhay E, Ramirez-Gonzalez G (2018) Convolutional neural network for bio-medical image segmentation with hardware acceleration. Cogn Syst Res 50(10–14):012003. https://doi.org/10.1016/J.COGSYS.2018.03.005
8. Zhao C, Han J, Jia Y, Fan L, Gou F (2018) Versatile framework for medical image processing and analysis with application to automatic bone age assessment. J Electr Comput Eng 2018. https://doi.org/10.1155/2018/2187247
9. Zhu S, Wang L, Duan S (2017) Memristive pulse coupled neural network with applications in medical image processing. Neurocomputing 227(149–157):012003. https://doi.org/10.1016/J.NEUCOM.2016.07.068
10. Cohen JP, Morrison P, Dao L, Roth K, Duong TQ, Ghassemi M (2020) COVID-19 image data collection: prospective predictions are the future, June 2020, Accessed 02 Sept 2021 [online]. Available http://arxiv.org/abs/2006.11988
11. Medical Definition of Idiopathic. https://www.medicinenet.com/idiopathic/definition.htm. Accessed 08 Sept 2021

Acute Leukemia Classification and Prediction in Blood Cells Using Convolution Neural Network

M. Shanmuga Sundari, M. Sudha Rani, and Kodumuri Bhargav Ram

Abstract Nowadays, human health is paramount to any other thing in the world. But health is affected due to many reasons. Doctors and scientists are constantly working to find solutions to health issues. The main issue is blood-related problems because blood is the foundation of our body. Cancers related to blood are very critical and cause human death. Leukemia is a kind of most cancers that arises inside the bone marrow and outcomes in an excessive wide variety of peculiar white blood cells. If acute leukemia cannot be treated in a short time, there is less time for humans to survive. It is important to detect cancer at an early stage and be able to treat it. It takes more time to cure so early detection is vital for treating cancers. In this research, machine learning is used to predict cancer cells in the blood. So, we used convolutional neural networks (CNN) to train the model and find cancer cells in the blood at an early stage. This research shows the prediction of blood cancer cells and displays the differences between the normal and cancer cells image using CNN classification process.

Keywords Acute leukemia · Convolution neural network · Lymphoid cells · Myeloid cells

1 Introduction

Leukemia [1] is also known as cancer of white blood cells. The white blood cells are a critical part of cells that will boost our immune level in the body. It is produced

M. Shanmuga Sundari (✉) · M. Sudha Rani · K. B. Ram
BVRIT HYDERABAD College of Engineering for Women, Hyderabad, Nizampet Rd, Hyderabad, Telangana, India
e-mail: sundari.m@bvrithyderabad.edu.in

M. Sudha Rani
e-mail: sudharani.m@bvrithyderabad.edu.in

K. B. Ram
e-mail: bhargavram.k@bvrithyderabad.edu.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_11

in the bone marrow although some types are produced in the lymph nodes spleen and thymus gland. The white blood cells are potent infection fighters protecting the body from invasion by bacteria, viruses, fungi, foreign substances, and abnormal cells. It grows normally and divides in an orderly manner as needed by the body. However, in the case of leukemia, white blood cells are unusually generated in the bone marrow that does not characterize like regular white blood cells. The abnormal white blood cells divide too quickly eventually crowding out normal cells. Leukemia is categorized into acute or chronic conditions based on how fast it progresses. The type of cells involved is lymphocytic or myelogenous [2]. In acute leukemia, the abnormal cells are immature. They increase rapidly making the bone marrow unable to produce healthy cells and the disease worsens quickly. Therefore, immediate and aggressive treatment is required at the earliest. These types are common across all age groups, affecting children to older people. The blood cells build up more slowly and take months or years to progress. Lymphocytic leukemia influences the lymphoid cells that affect the lymphatic tissues and immune system.

2 Related Works

CMOS sensing circuits used to predict breast cancer [3] and RNA-195 detection are also discussed in this paper. TML and DL techniques in MIA [4] are used for identifying leukocyte classification. Computer aided-detection (CADx) was used to bet better performance to diagnose blood smear problems. Rapid immunoanalysis of physiological fluids [5] was identified using the signal-to-noise ratio of the biosensor.

The conventional machine learning methods [6] are used for predicting cancers in the research areas. Author used many machine learning algorithms for this research. The author states that the DCNN model's performance is increased using CNN techniques. Kidney failure [7] was also proved using glomerular filtration rate, and kidney-inspired algorithm was used to find the kidney capacity of usage in the duration of the failure. The photoacoustic imaging [8] which is in the proposed system was implemented by two cervical tissues with the image of mimicking phantoms with human blood and graphiterods as inclusions inside it. Prostate-specific antigen (PSA) blood level [9] is an important value that is elevated mostly in men with prostate cancer, and localized fluorescence (XRF) [10] is well used to compute tomography (XFCT) with nanoparticles (NPs). The error backpropagation neural network (BPNN) [11] algorithm is used, and spectrum is applied to detect cancer in medical images [12]. Machine learning algorithms are used to predict many medical diseases.

3 Proposed System

Blood cells are having different types of cells:

- Red cells
- White cells
- Platelets.

Red blood cells are boosting the oxygen transportation from the lungs to the other tissues. White blood cells are responsible for fighting diseases and reducing infections. Platelets [13] are used to clot the blood vessels when it is hurt. In our body, we have a ratio of 1000 red cells: 1 white cell.

White cells turns into leukemia, and the types are:

- Lymphoid cells
- Myeloid cells.

Leukemia caused by myeloid cells is stated as myelogenous or myeloid leukemia. Based on the intensity of leukemia, this is classified into two different ways: chronic or acute. Cells are grouped according to the speed of growing the cells inside the body. If the leukemia is very chronic, the cells will end in affecting the body and disease get worse.

4 Dataset Description

In our research, the dataset [14] has images of patients with leukemia problems. The images are multiple myeloma representation. We have 4961 training images. Out of that, we have 2483 healthy patients and 2478 images are blood cancer. The resolution is 320 * 240. The proposed system is displayed in Fig. 1.

4.1 Data Augmentation

Our research dataset is focused on rotating and extracting edges from the images. The shuffled images are divided into two categories. They are training and testing sets. There are many characteristics that are considered to make an evaluation that are varying with different attributes like sizes, positions, clarity, and lighting conditions helping to evaluate.

4.2 Feature Selection

Feature selection is an important process in deep learning to get better results. Hence, feature selection is playing a vital role in the entire process. The way of selecting efficient features is called feature selection [15]. By this, we can reduce overfitting and increase under-fitting [16].

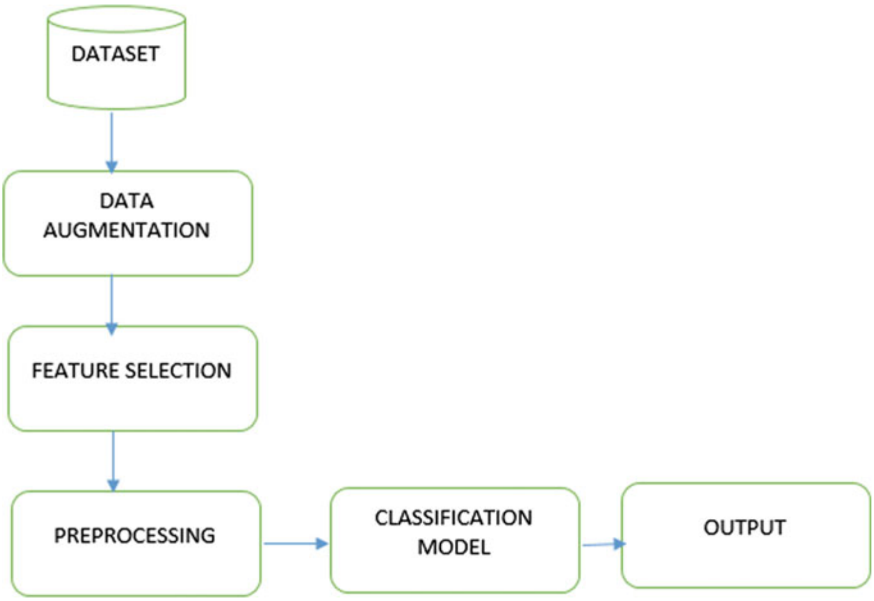


Fig. 1 Proposed system

In this research, we use univariate feature selection and which will help us to get the target factors. In our research, we use *K*-specific features and classes for finding the chi square. This test is done between the independence of two cells. Consider the observation count *C* and prediction count *P* given the two variables. We select the high dependent cell values in between the different cells. When the observed and predicted values are closer, it will give a small chi square value.

So, the high value of chi square will define independence hypothesis. Hence, model is building with dependent on higher the value of chi square which is responsible for feature selection. The chi-squared statistical test is shown in Eq. (1).

$$x^2 = \frac{(c - p)^2}{p} \tag{1}$$

where *C* represents the observation count of the class, *P* represents the number of expected observations of the class when there is no relation between the feature and the response.

4.3 CNN Techniques

CNN is proposed for segmentation to differentiate with color-based clustered to get nucleus region and stained blood smear images in cytoplasm area. SVM classifiers [17] are best to get satisfactory results. The system is built to detect automatically white cells from the blood culture test. Based on that WBCs are classified into five types: eosinophil, basophil, neutrophil, monocyte, and lymphocyte.

CNN is the convolution model; Fig. 2 gives automatic feature extractors and finds the resolution with pixel vector.

In a convolution network, the first operation is feature extraction from the image. Here we are using the $k \times k$ matrix to get the particular regions from the image using an activation map.After getting the activation region, we discord the error value using equation 2. The image is moved like a stride pattern. Operation for our data image size is $x \times y$ and the size of v with padding value d. v is the vector value for feature. S is constant value. I is the negligible error value.

$$(x - v + 2d) = s + I \times (y - v + 2d)/s + 1 \tag{2}$$

Filter depth is also based on the image resolution and pattern. In this model, we use convolution layers and a pooling layer.

Equation (3) gives the activation map after formulating the values from convolution operation.

$$\sigma(z)_i = \frac{e^{zi}}{\sum_{j=1}^{k} e^{zj}} \tag{3}$$

for $i\,D\,1;:::;K$ *and* $z\,D\,(z1;:::;zK)\,2 < A$, where A represents the input elements vector z and zi derived value from input vector z. Equation (4) gives the final feature selection function.

$$A[x, y] = \sum_{i}^{(I*f)} \sum_{j}^{[x,y]} I[i, j] f[x - i, y - i] \tag{4}$$

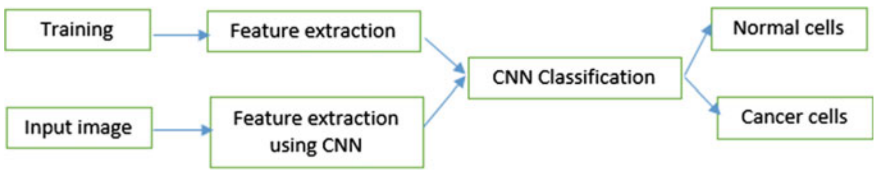


Fig. 2 Block diagram

4.4 Working of CNN

The CNN has a convolutional layer, pooling layer, and connected layer. Based on the RGB color image, the first six layers are considered convolution layers. The two layers apply 16 of 3 * 3 filters. The remaining layers apply 32 of the matrix (3 * 3) filters in the image. The eighth layer is considered a flatten layer. It is responsible for multidimensional arrays. The flatten array is 4900 in size. The layer is connected from 4900 input values and 64 output attributes. The tenth layer is helpful to reduce overfitting. The eleventh layer and last layer are mapped with input values to different class labels.

5 Results and Analysis

This research shows the detection of cancer cells using CNN that gives the better result with compare other techniques. With respect to dataset, we can diagnose errors in higher range using large dataset. The result analysis shows the detection of cancer cells accurately and is shown in Fig. 3 with the loss and accuracy graph.

We did 20 series and observed clearly to decrease the loss in the iteration. The loss gives the accuracy of our model. We tried to minimize the loss function and get the confusion matrix shown in Figs. 3 and 4. In this, 379 cases fall into the true–true case in the confusion matrix.

Figure 5 displays the accuracy value of 0.78044. The research shows the accuracy for predicting the image of a cancer cell or not.

Figure 6 shows the normal blood cells and cancer cells pictures. The difference between the actual and cancer cells in the blood is shown in this image.

```
Epoch 00008: ReduceLROnPlateau reducing learning rate to 4.999999858590343e-11.

Epoch 00008: val_acc did not improve from 0.78205
Epoch 9/10
5216/5216 [==============================] - 7s 1ms/step - loss: 0.2243 - acc: 0.9085 - val_loss: 0.4974 - val_acc: 0.7796

Epoch 00009: ReduceLROnPlateau reducing learning rate to 4.999999719812465e-12.

Epoch 00009: val_acc did not improve from 0.78205
Epoch 10/10
5216/5216 [==============================] - 7s 1ms/step - loss: 0.2242 - acc: 0.9088 - val_loss: 0.4974 - val_acc: 0.7796

Epoch 00010: ReduceLROnPlateau reducing learning rate to 4.999999546340118e-13.

Epoch 00010: val_acc did not improve from 0.78205
```

Fig. 3 Loss value of blood cancer

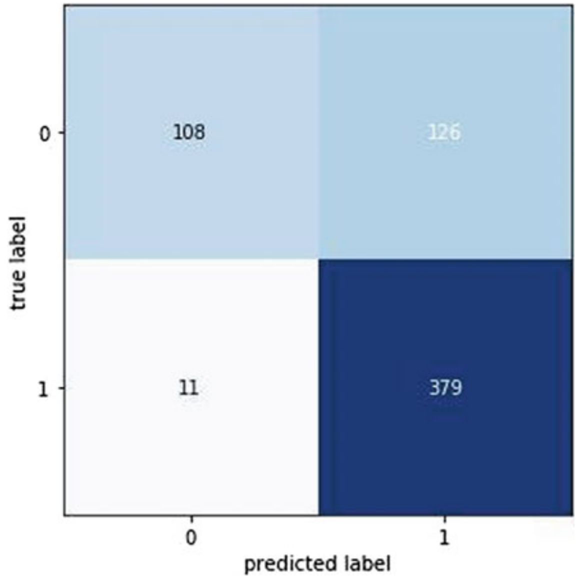


Fig. 4 Confusion matrix

```
In [16]: #PRECISION = (TP/(TP+FP))
         379/(379+126)

Out[16]: 0.7504950495049505

In [17]: #RECALL = (TP/(TP+FN))
         379 / (379 + 11)

Out[17]: 0.9717948717948718

In [18]: #ACCURACY = (TP+TN)/(TP+TN+FP+FN)
         (379+108)/(379+108+126+11)

Out[18]: 0.780448717948718
```

Fig. 5 Accuracy value for predicting blood cancer

6 Conclusion

The proposed model is to find acute leukemia in blood cells using a convolution neural network. This model uses the images as input and predicts the cancer cells in the blood cells. The accuracy of prediction shows a value of 78.04%. A baseline model is used to compare the accuracy of the images with the CNN model. Our dataset

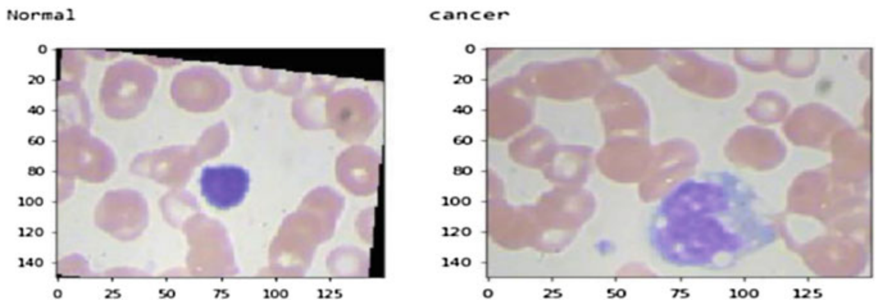


Fig. 6 Cancer cell and normal cell image

supports the highest accuracy of 78.04%. By this, we can predict acute leukemia in the blood cells that can help in the treatment. The patients can get medication at an early stage so that we can save their life. In the future, we will implement models with different machine learning algorithms and predict the outcomes.

References

1. Kantarjian H et al (2021) Acute myeloid leukemia: current progress and future directions. Blood Cancer J 11(2):1–25
2. Kumar D et al (2020) Automatic detection of white blood cancer from bone marrow microscopic images using convolutional neural networks. IEEE Access 8:142521–142531. https://doi.org/10.1109/ACCESS.2020.3012292
3. Kuo Y-H, Chen Y-S, Huang P-C, Lee G-B (2020) A CMOS-based capacitive biosensor for detection of a breast cancer MicroRNA biomarker. IEEE Open J Nanotechnol 1:157–162. https://doi.org/10.1109/ojnano.2020.3035349
4. Punj S, Sidhu D, Bhattacharya D, Wang M, Wong PK (2020) An electrochemical biosensor platform for rapid immunoanalysis of physiological fluids. IEEE Open J Nanotechnol 1:31–37. https://doi.org/10.1109/ojnano.2020.2997296
5. Li G, Li H (2021) Linear model selection and regularization for serum prostate-specific antigen prediction of patients with prostate cancer using R. IEEE Access 9:97591–97602. https://doi.org/10.1109/ACCESS.2021.3095914
6. Lakshmi L, Purushotham Reddy M, Praveen A, Suniha KVN (2020) Identification of diabetes with recursive partitioning algorithm using machine learning. Int J Emerg Technol 11(3)
7. Ahlawat S, Batra V, Banerjee S, Saha J, Garg AK (2019) Hand gesture recognition using convolutional neural network. In: International conference on innovative computing and communications, pp 179–186
8. Basij M, Karpiouk A, Winer I, Emelianov S, Mehrmohammadi M (2021) Dual-illumination ultrasound/ photoacoustic system for cervical cancer imaging. IEEE Photon J 13(1). https://doi.org/10.1109/JPHOT.2020.3043685
9. Shinde SA, Rajeswari PR (2018) Intelligent health risk prediction systems using machine learning: a review. Int J Eng Technol. https://doi.org/10.14419/ijet.v7i3.12654
10. Shaker K et al (2020) Longitudinal in-vivo X-ray fluorescence computed tomography with molybdenum nanoparticles. IEEE Trans Med Imaging 39(12):3910–3919. https://doi.org/10.1109/TMI.2020.3007165

11. Padmaja B, Prasad VVR, Sunitha KVN, Reddy NCS, Anil CH (2019) Detectstress: a novel stress detection system based on smartphone and wireless physical activity tracker. In: Advances in intelligent systems and computing, vol 815. https://doi.org/10.1007/978-981-13-1580-0_7
12. Sundari MS, Nayak RK (2020) Process mining in healthcare systems: a critical review and its future. Int J Emerg Trends Eng Res 8(9):5197–5208. https://doi.org/10.30534/ijeter/2020/50892020
13. Kumari PIC, Gayathri P, Rajesh N, Umar S, Sekhar GC, Abdul AM (2016) Designing of medical processor unit for intelligent network-based medical usage. Indones J Electr Eng Comput Sci. https://doi.org/10.11591/ijeecs.v4.i3.pp532-537
14. https://www.kaggle.com/andrewmvd/leukemia-classification
15. Wang S, Wang S, Zhang S, Fan F, He G (2020) Research on recognition of medical image detection based on neural network. IEEE Access 8:94947–94955. https://doi.org/10.1109/ACCESS.2020.2995466
16. Abdullah S, Jaddi NS, Jaddi NS (2020) Dual kidney-inspired algorithm for water quality prediction and cancer detection. IEEE Access 8:109807–109820. https://doi.org/10.1109/ACCESS.2020.3001685
17. Khan S, Sajjad M, Hussain T, Ullah A, Imran AS (2021) A review on traditional machine learning and deep learning models for WBCs classification in blood smear images. IEEE Access 9:10657–10673. https://doi.org/10.1109/ACCESS.2020.3048172

Process-Based Multi-level Homogeneous Ensemble Predictive Model for Analysing Student's Academic Performance

Mukesh Kumar and Amar Jeet Singh

Abstract The aim of this study is to undertake an empirical inquiry and comparison of the effectiveness of various classifiers with ensembles classifiers in the prediction of student academic performance. A single classifier algorithm will be compared against the performance and efficiency of ensemble classifiers. Reducing student attrition is a major problem for educational institutions all over the world. The search for solutions to increase student retention and graduation rates continues for educators. This is only possible if at-risk students are identified and intervened with as soon as possible. However, the majority of regularly used prediction models are inefficient and inaccurate as a result of inherent classifier limitations and the inclusion of insignificant inputs in their calculations. The majority of data mining and machine learning researcher focused on developing an algorithm that can extract useful information from massive amounts of data after being processed by a computer. The most difficult problem in predictive modelling is identifying the most effective prediction algorithms that are also accurate enough to be useful. Therefore, a multi-level homogeneous ensemble predictive (MLHoEP) model is designed, which uses the different techniques of data mining like feature selection, ensemble learning techniques like boosting and bagging. Seven distinct machine learning algorithms were used on this model to predict and analyse the academic performance of the students. The performance of the classification algorithms in terms of prediction was evaluated using k-fold cross-validation. The study contributes to the body of knowledge by suggesting the development of homogeneous classifiers that may be used to accurately predict students' academic success. It also proposes the construction of homogeneous classifiers, which may be deployed for accurate student performance prediction, in order to provide a better explanation for the poor performance prediction. As a result of this research, it has been demonstrated that the technique of applying homogeneous ensemble approaches is incredibly efficient and accurate in terms of predicting student performance and assisting in identifying students, who are in danger of dropping out of school. The study compared the accuracy and efficiency of single classifiers to ensembles of classifiers in terms of performance. It was discovered in the

M. Kumar (✉) · A. Jeet Singh
Department of Computer Science, Himachal Pradesh University, Shimla, India
e-mail: mukesh.kumarphd2014@gmail.com

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_12

research that a homogeneous model with excellent accuracy and efficiency might be developed for anticipating student performance. These key problems have been successfully addressed by the findings of this research study: Which characteristics of students are the most effective predictors of academic performance? How accurate are approaches such as bagging and boosting ensembles for predicting student academic performance? The approach offered in this study will aid educational administrators and policymakers in designing new policies and curriculum-linked to student retention in higher education. This research can also aid in the identification of students who are at risk of dropping out of school early, providing for timely intervention and support. Prospective research will examine the creation and implementation of an automated prediction system known as the students' academic performance forecast framework, which will collect data from students via online submission and produce a prediction result for their academic performance.

Keywords Educational data mining · Ensemble learning · Multilayer perceptron · Random forest · Naïve Bayes · Correlation attribute evaluation · Information gain · Gain ratio

1 Introduction

Ensemble learning is frequently used to average the predictions of multiple classification models in order to obtain a more accurate forecast. This strategy is used to forecast the outcomes of insignificant classifier models drawn from a variety of input spaces. It is simply a technique for integrating the outputs of multiple models to obtain a more accurate result [1]. This is one of the simplest and most cost-effective strategies for enhancing the accuracy of your model's prediction outcomes. The majority of real-world applications employ some form of ensemble approach to improve the prediction model's performance. After the Netflix challenge, where all of the winning teams used ensembles of numerous inconsequential models to win, these ensemble learning techniques became increasingly prevalent. Netflix's primary goal with this challenge was to develop a new recommendation system that would allow users to suggest new films [15]. Additionally, these strategies are applied to deep learning. In data mining, certain classification algorithms, such as the random forests technique, are implemented using ensembles. The simplest technique for training a random forest algorithm is to train multiple decision trees on distinct subsets of the dataset using different feature subsets and then average the results [3].

Bagging and boosting are two further examples of assembly. Bagging is a technique that entails running numerous models on distinct sets of input samples and then averaging the results. Bagging is advantageous when the objective is to reduce variation while maintaining the same bias [6]. When used on an overfitted model with low bias and a large variation, bagging is beneficial. It is ineffective in cases where models exhibit a high degree of bias. Essentially, ensemble learning is group learning. Essentially, ensemble learning is a technique in which we train a large number of

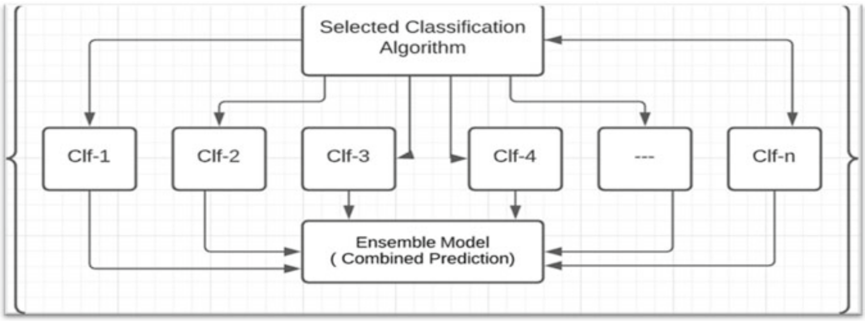


Fig. 1 Representation of ensemble learning methods

unimportant models and then integrate their predictions to arrive at a conclusion [2]. Combining the forecasts is a procedure that is determined by the models trained. If the models are homogeneous, that is, if all trained models utilise the same algorithm, such as decision trees, we can apply either bagging or boosting (Fig. 1).

These are the most often utilised ensemble learning approaches. If the trainers are diverse and a combination of multiple algorithms is utilised, such as decision trees, logistic regression, and so on, meta-learning can be applied. In this example, on top of all the predictions, you train another model that determines the final prediction [11]. Assume the learners generate class probabilities using a combination of 100 decision trees and logistic regression. You will end up with 100 values for each training instance; you can then train another model to predict the real outcome using these 100 values.

2 Literature Survey

During this study, we came across different research papers related to implementation of ensemble learning algorithms and see how these algorithms improve the prediction result of different classifier model. Different researcher groups of education data mining communities are working in different areas of education and its development [10]. Their research is focused on to find the effect of different student's attributes on academic performance, predict the academic performance of the student's, and predict the placement of the student [20]. In article [16], a survey of the literature is presented and certain theoretical methods are implemented in order to forecast student performance. For example, she discovered and compared the accuracy of Naive Bayes, Neural Network, and Decision Tree to predict students' cumulative grade point average (CGPA),students' demographics, high school, and study and social network attributes as the most critical factors in whether students pass or fail their studies [4]. The accuracy of naive Bayes is higher than that of neural networks and decision trees because it uses attributes that are more significant to forecast.

Educational data mining (EDM) is an interdisciplinary field that is concerned with the creation of methods to analyse a variety of unique data in the education area, with the goal of better understanding students' requirements and determining appropriate learning approaches [13]. In general, EDM is used to foresee difficulties in order to improve the quality of both student performance and the teaching–learning process [12], as well as the overall teaching–learning process. Due to the large amount of data in the educational dataset, it is concerned with how to adapt data mining methods and identify patterns, which are normally highly difficult problems to solve [14]. In order to identify datasets, data mining as a decision-making tool has been aided by a variety of approaches, including statistical models, mathematical methods, and machine learning algorithms [5]. Yet another piece of research, paper [18], examines numerous and relevant data mining approaches for classification in prediction, primarily for the purpose of determining the most important aspects of student performance forecasts. Using the random forest and J48 classification models, it is possible to forecast student achievement and to identify the most significant factors that influence it, such as study time spent, academic year attended, and parental education. In this paper [19], artificial neural networks, decision trees, and Bayesian networks were utilised to detect dropouts in order to investigate a large number of probable factors. Tan found two attribute variables as test inputs while doing empirical research on a dataset containing 3.59 million student records from an online training programme. These attribute variables were student characteristics and academic performance. As a result, the decision tree method was more exact in demonstrating that those variables are effectively used as key components in the prediction of student dropouts than before. As demonstrated in this work [9], Marquez presented a novel strategy for optimising the accuracy of predictive modelling, which he named modified interpretable classification rule mining, to improve predictive modelling accuracy. Marquez conducted an experiment in 419 schools to determine the elements that contribute to student dropouts. Six steps of evaluation were carried out, with a total of 670 students providing 60 different factors. As a result, modified classification rule mining is more accurate than JRip in terms of accuracy. Predictive modelling issues currently include the effectiveness and accuracy of various prediction models, which are mostly caused by insufficient variables in the basic classifier in most cases. In a related study [8], decision trees, naive Bayes, KNNs, and artificial neural networks were used to construct a predictive student dropout model and to adopt ensemble clustering based on students' demographic information, academic performance, and enrolment history, respectively. The accuracy of prediction models can be improved by using an experiment-verified ensemble approach to transform original data into a new form. Another similar study, as stated in [7], explored and investigated the ensemble technique, which was found to be effective in reducing errors and increasing the accuracy of student performance prediction. Below are some takeaway from this literature review:

Student's Attributes Which Effect the Academic Performance Prediction: There are so many student attributes, which effect the academic performance of any student from academic, family, institutional, social or personal attributes. Which attribute effect the student performance most is a matter of research for each and every researcher in the field of educational data mining. But it all depend upon the output you want from your predictive model. Some researcher wants to predict the student dropout status, some wants to predict the student placement, and some wants to predict the final grade of the student and many more. So, in the literature, there is no fixed attributes which we can say totally effect the overall performance of the student in school or any other institution during their study. But surely, we found that the categories of student attributes which overall play some role in predicting the academic performance of the students and these are academic attributes, family attributes, and institutional attributes.

Classification algorithms mostly used to predict academic performance student's: An important task for predicting academic performance of student's is to develop a superior classifier model by using classification algorithms. There are lots of families of classification algorithms which are built in the past by different researcher. At the time literature review, we came across such different algorithms which gave different types of accuracy on to the selected datasets for predicting academic performance of the students.

We want to say that in educational data mining, to improve the overall prediction accuracy of any classification algorithms, we have some ensemble learning techniques and these techniques are bagging, boosting, and random subspace. So, we proceed with our work by taking below-mentioned questions in our mind; first is how classification algorithms are valued for predicting academic performance of students and second is how classification algorithms performance is improved by using different ensemble learning techniques.

3 Materials and Methods

3.1 Data Description

This dataset pertains to student achievement in secondary education at two Portuguese educational institutions [17]. Among the information gathered from the students were student grades as well as demographic, social, and school-related attributes. The information was obtained through school reports and questionnaires. On the basis of performance in two independent subjects: mathematics (mat) and the Portuguese language (por), two datasets are offered. Cortez and Silva [7] used the two datasets to simulate classification and regression tasks that were either binary or five-level classification or regression tasks. One thing to keep in mind is that the target attribute G3 has a high association with the other two traits, G2 and G1. This is due to the fact that G3 is the final year grade (which is delivered at the end of the

Table 1 New class level assigned to the dataset

Range of initial class given in the dataset	New cluster number assigned to the class level
Grade between 16 and 20	Class-A
Grade between 14 and 15	Class-B
Grade between 12 and 13	Class-C
Grade between 10 and 11	Class-D
Grade less or equal to 9	Class-F

third period), whereas G1 and G2 correspond to the first and second period grades, respectively. Even though it is more difficult to anticipate G3 without first predicting G2 and then G1, such predictions are far more valuable. The desired output class initially has a range of 0–20, and there are 21 clusters. This is an unreasonable option for the classification task, as it makes classification extremely difficult, especially given the small number of instances available. In the given dataset, G1, G2, and G3 and the grade obtained by different students and for better result we find the final grade of the student by find the average of all grades and create a new attribute named as "total grade". As a result, I have assigned a group of clusters to a few class levels denoted by the letters A, B, C, D, and F in Table 1.

3.2 Classification Algorithm Used

Classification is a data mining technique that classifies the elements in a dataset. The objective of classification is to accurately anticipate the target class for each occurrence of data. For instance, a classification model could be used to classify loan applicants into three categories based on their credit risk: low, medium, and high. Several classification techniques have been chosen for implementation, as follows:

Naïve Bayes: Naive Bayes is a model which is based on Bayes' theorem and makes several fiercely independent assumptions. It forecasts the probability that a particular instance in a dataset belongs to a specific class. It is presumed that the prevalence of a feature in a class is unrelated to the presence of any other characteristic, i.e. that all features contribute independently in calculating the probability of data classification. This model is advantageous for very huge datasets and is simple to implement.

Random Forest: It is an ensemble method that combines various decision trees and a bagging technique. Bagging is the process of training each decision tree using a portion of the original dataset obtained through sampling and replacement. The final class is determined by conducting a majority vote on the outcome of all decision trees. It is an extremely efficient and effective technique when dealing with enormous datasets.

Decision Tree: The decision tree algorithm, also known as induction of decision trees, is a technique that is used in statistics, data mining, and machine learning to do predictive modelling and classification. It progresses from observations of an object's attributes to judgments about the item's desired value through the use of a decision tree.

Multilayer Perceptron: This is a sort of feedforward neural network that has multiple layers (ANN). Backpropagation is a supervised learning strategy that is used to train the algorithm. A MLP differs from a linear perceptron in that it has many layers and nonlinear activation, whereas a linear perceptron has only one layer. It has the capability of separating data that are not linearly separable, among other things.

Decision Table: Specific attributes are considered during the learning process of this classifier. This is accomplished by computing the table's cross-validation performance in various subsets of attributes and picking the subsets that performs the best. The cross-validation error is calculated by changing the class counts associated with each dataset entry, as the table structure remains constant, when instances are added or deleted. Typically, the feature space is searched using a best-first search method.

JRip: This class provides a learner for propositional rules, which can be used to automate the learning process. This approach was developed by William W. Cohen as an acceptable algorithm for the IREP. It employs a technique known as repeated incremental pruning in order to reduce error rates (RIPPER).

Logistic Regression: When there are many explanatory factors, logistic regression is used to compute the odds ratio. When there are multiple explanatory variables, logistic regression is used to calculate the odds ratio. The approach is quite similar to multiple linear regression. However, the response variable is a binomial distribution instead of a linear distribution. The outcome is defined as the effect of each variable on the odds ratio of the observed occurrence.

3.3 Ensemble Learning Method Used

When using ensemble learning, numerous data mining models are combined to create more efficient and effective learning algorithms, which ultimately improves the accuracy of any model's prediction output. This strategy combines numerous weak learners in order to increase the accuracy of our predictive models. In ensemble models, the decision tree is frequently chosen as the weak learner, and this is because of its simplicity. The core concept behind ensemble learning is that it involves training a large number of inconsequential models and then combining the predictions to get a conclusion. The strategy used to combine the predictions is determined by the models that were used in the training process. If the models are homogeneous, meaning that all of the trained models use the same algorithm, such as the decision tree, then you can use either bagging or boosting to optimise the performance of the model. Gradient boosters, which are ensemble models, have grown increasingly popular.

Bagging Ensemble Learning: Bootstrap aggregation is the technical term for bagging. By producing some additional data for training from your original dataset, utilising combinations with repetitions to build multisets of the same size as your original data, it is possible to reduce variation in the outcome of your prediction. You will not improve the predictive accuracy of your model by increasing the size of your training set, but you will minimise the variance of your model, narrowing the forecast to the most likely outcome.

Boosting Ensemble Learning: It is a technique for creating a collection of predictive models that are used in conjunction with other techniques. Predictive models are taught sequentially using this technique, with early models fitting simple models to the data and then analysing the data for errors before learning more complex models. Remember that bagging requires each model to be run independently and then the outputs be aggregated at the end without giving any preference to any particular model.

3.4 Correlation Attribute Evaluator (CAE)

Methods for feature selection try to minimise the number of input variables to those that are deemed to be most beneficial in predicting the target variable. The purpose of feature selection is to exclude uninformative or redundant predictors from the model. Calculate the value of an attribute by calculating the correlation (Pearson's correlation coefficient) between it and the class. Nominal qualities are analysed value by value, with each value acting as an indicator. A weighted average is used to determine the overall correlation for a nominal property.

4 Proposed Multi-level Homogeneous Ensemble Predictive Model

In the below-mentioned Fig. 2, we demonstrate the working architecture of the proposed machine learning algorithms in conjunction with other important application algorithms of machine learning like feature selection (FS) and ensemble learning (EL) algorithms along with *k*-fold cross-validation as a testing method. At the start, first we need to select a dataset which is related to academic performance of the students with different features (independent and dependent). During the pre-processing phase, we need to remove all types of discrepancy be there in the dataset during data collection. Now, it is time to test our dataset in two different modes; first mode is to test our dataset with all the features present in it, and second

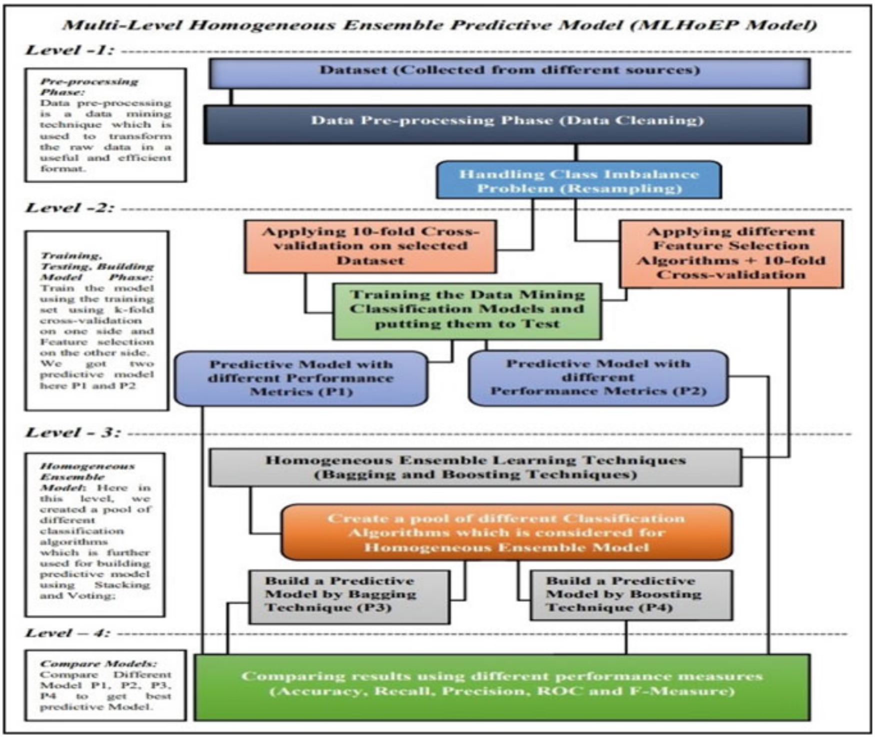


Fig. 2 Design of multi-level homogeneous ensemble predictive model

mode is to select some of the features with feature selection (FS) algorithm. Here, only correlation attribute evaluator (CAE) is used to implement FS and only top ten attribute are selected to find the accuracy of the classification algorithms. Now, move to next step where we need to select the testing mode along with the classification's algorithms for the implementation. Now, it is time to select which ensemble learning algorithms need to be implemented to test the classifications algorithms.

A unique technique is called the multi-level homogeneous ensemble predictive model (MLHoEP model). As we saw throughout the literature review step, the majority of authors relied solely on data to arrive at the best outcome. However, in our MLHoEP model, we outlined a process that must be followed whenever homogeneous ensemble predictive modelling is used. In the MLHoEP model, we divide our predictive process into distinct levels, and each level will tackle its own set of problems. The following is a block diagram of the MLHoEP model:

Level-1: Prior to progressing to the next level, manage missing values (by mean and me¬dian), outliers, and class imbalances in the dataset (Resampling Method).
Pseudo Code for level-1 in MLHoEP Implementation
Level-1: Data Pre-processing Phase

```
# Here, feature domain is {fl, f2, f3,…,fn}
# Handling Missing Value by mean()
1: Replace_Missing_Value_Mean(dataset)
2: return dataset [‘fl’, ‘f2’, ‘f3’, …, ‘fn’]. replace (‘O’, mean())
# Handling Missing Value by median()
3: Replace_Missing_Value_Median(dataset)
4: return dataset [‘f4’, ‘f5’, …, ‘fn’]. replace (‘O’, median)))
5: Train_Test_Data_Split( diabetes)
# Handling imbalance problem by Oversampling
6: dataset minority oversampled(dataset)
7: retrun resample(l, replace = True, nsamples = majority class instance)
8: dataset = pd.concat([0, datasetminorityoversampled])
# Handling unbalance problem by Undersampling
9: dataset majority undersampled(dataset)
10: retrun resample (1, replace = True, n samples = minority class instance)
11: dataset = pd.concat([0, dataset_ majority undersampled])
```

Level-2: At this level, various classification methods are implemented and verified for accuracy (both with the complete dataset and with feature selection). Here, we have two clas¬sifiers, PI and P2.

Pseudo Code for level-2 in MLHoEP Implementation

Level-2: Training, Testing, Building Model Phase #Building Predictive Model (PI)

```
#Splitting dataset into Training and Testing dataset
12: Traing_SpIit, Testing_Split = split (dataset_feature_space, dataset_class_level)
13: return TraingSplit, Testing Split
#Applying k-fold cross validation on selected dataset
14: CV = k_fold_cross_validation (n_splits=10. random_state = 1, shuffle = True)
#Building Different Classifiers
15: Model-1: NBModel(Traing Split, Trainglabel, Testing Split)
16: Model-2: RFModel(Traing Split, Traing label, Testing Split)
17: Model-3: DTMndel(Traing Split, Traing label, Testing Split)
18: Model-4: MT PModel(Traing_Spl it. Traing_label, Testing_Split)
19: Model-5: DTModel(Traing Split, Traing label, Testing Split)
20: Model-6: JRipModel(Traing Split. Traing label, Testing Split)
21: Model-7: LRModel(Traing_Split, Traing label, Testing_Split)
#Building Predictive Model(P2)
#Applying different Feature Selection Algorithms
22: impattribute = model.CAE
23: for i, v in enumerate (imp_attribute):
24: Result v
25: Select top m feature according to your problem
#Applying k-fold cross validation on selected dataset
```

26: CV = k_fold_cross_validation (n_splits=10, random_state=l, shuffle=True)
#Applying different Feature Selection Algorithms + k-fold cross validation
27: Model-1: NBModel (Traing Split, Traing label, Testing Split)
28: Model-2: RFModel (TraingSplit, Traing label, Testing Split)
29: Model-3: DTModel (Traing_Split, Traing_label, Testing_Split)
30: Model-4: MLPModel (Traing Split, Traing label, Testing Split)
31: Model-5: DTModel (Traing Split, Traing label, Testing Split)
32: Model-6: JRipModel (Traing Split, Traing label, Testing Split)
33: Model-7: (Traing_Split,Traing_label, Testing_Split)

Level 3: In this section, we develop a pool of diverse categorization methods that must be considered while constructing a homogenous ensemble model. Indeed, we are evaluating only those algorithms that have been selected for implementation at the level-2 level. We obtain P3 and P4 predictive models from this level.

Pseudo Code for level-3 in MLHoEP Implementation

Level-3: Homogeneous Ensemble Model

#Building Predictive Model (P3)
#Splitting dataset into Training and Testing dataset
34: Traing split, Testingsplit = split (dataset feature space, dataset class level)
35: return Traing split, Testing split
#Applying K-fold Cross-validation on selected Dataset
35: CV = k_fold_cross_validation (n_splits = 10, random_state = l, shuffle = True)
36: pool_of_classification_Model (Modell, Model2, Model7)
37: Compare accuracy of each model in the pool with the highest model achieved
38: Ensemble_Model (TraingSplit, Trainglabel, TestingSplit)
39: ModellBagging.fit (Traing Split, Traininglabel)
40: Model2_Bagging.fit (Traing Split, Training label)
41: Model3_Bagging.fit (Traing_Split, Training label)
42: Model4_Bagging.fit (Traing_Split, Training_label)
43: Model5_Bagging.fit (Traing_Split, Training_label)
44: Model6_Bagging.fit (Traing Split, Training label)
45: Model7_Bagging.fit (Traing Split, Training label)
#Building Predictive Model (P4)
#Splitting dataset into Training and Testing dataset
46: Traing_split, Testing_split = split (dataset_feature_space. dataset_class_level)
47: rehtrn Traingsplit. Testingsplit
#Applying k-fold cross validation on selected dataset
48: CV = kfoldcrossvalidation (n_splits = 10, random_state = l, shuffle = True)
#Building Boosting Ensemble Model
49: pool_of_classification_Model (Modell, Model2, …, Model7)

50: Compare accuracy of each model in the pool with the highest model achieved
51: EnsembleModel (TraingSplit, Traing label, TestingSplit)
52: Modell_Boosting.fit (Traing Split, Training label)
53: Model2_Boosting.fit (Traing_Split, Training label)
54: Model3_Boosting.fit (Traing_Split, Training_label)
55: Model4_Boosting.fit (Traing_Split, Training_label)
56: Model5_Boosting.fit (Traing Split, Training label)
57: Model6_Boosting.fit (Traing Split, Training label)
58: Model7_Boosting.fit (Traing_Split, Training label)

Level 4: Compare the predictive models (PI, P2, P3, P4) for better result on perfor¬mance metric.

All the necessary requirement are now set to implement the above-mentioned hybrid classification algorithms with the help of feature selection and feature selection algorithms. At the end, we need to compare all the implemented algorithms with each other to find the best on which gave use the maximum accuracy in prediction the result.

5 Implementation of the Proposed MLHoEP Model

Model Construction for the Standard Classifier: Numerous classification techniques were chosen and used to the dataset of student performance. We implement the following classification algorithms: naïve Bayes, random forest, J48, multilayer perceptron, decision table, JRip, and logistic regression. The table below summarises the implementation results of various categorisation algorithms using ten cross-validation (*k*-fold cross-validation) approaches. Our dataset is a balanced dataset with nearly equal distribution of data across five distinct classifications. According to Table 2, *decision tree* classification method had the greatest accuracy of 96.76% when

Table 2 Accuracy achieved by classification algorithm with all features of dataset

Classification algorithm	Accuracy (%)	MAE	Precision	Recall
Naïve Bayes + *k*-fold cross-validation	86.59	0.063	0.870	0.866
Random forest + *k*-fold cross-validation	92.14	0.118	0.924	0.921
Decision tree + *k*-fold cross-validation	*96.76*	0.017	0.968	0.968
Multilayer perceptron + *k*-fold cross-validation	84.59%	0.078	0.846	0.846
Decision table + *k*-fold cross-validation	90.13	0.146	0.909	0.901
JRip + *k*-fold cross-validation	96.14	0.024	0.962	0.961
Logistic regression + *k*-fold cross-validation	87.51	0.049	0.877	0.875

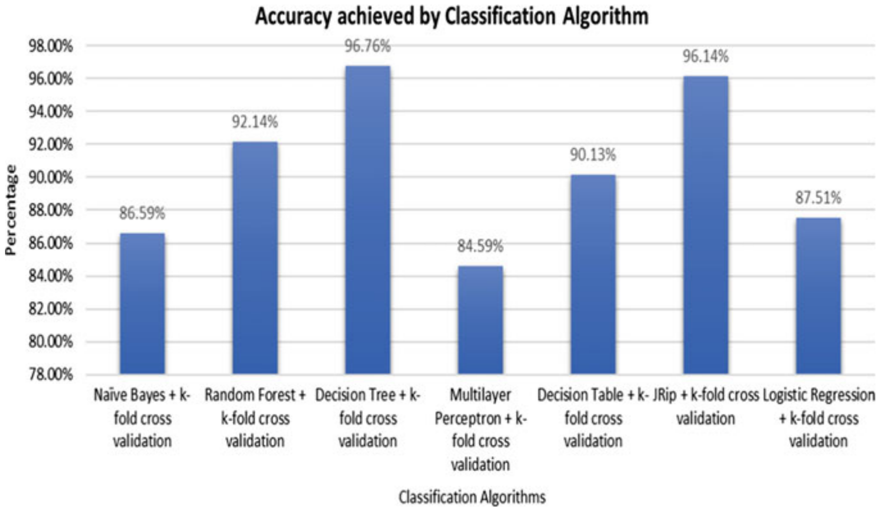


Fig.3 Graphical representation of accuracy level of classification algorithms

compared to other classification algorithms such as naive Bayes, random forest, decision table, multilayer perceptron, JRip, and logistic regression. As shown, the multilayer perceptron method achieves the lowest accuracy of 84.59%. Random forest and JRip algorithms also obtained an acceptable level of accuracy, at 92.14% and 96.14%, respectively. To implement these algorithms, all of the dataset's attributes (up to 32) are considered. Other performance metrics such as mean absolute error (MAE), precision, recall, ROC area, and *F*-measure are also considered in this table. As our dataset contains no outliers, we will use accuracy as our primary parameter for evaluating our classifier's effectiveness.

Classification algorithm accuracy is defined as the total number of correct predictions divided by the total number of predictions made by an algorithm for a given dataset. Figure 3 shows the graphical representation of the above-mentioned implementation of the classification algorithms with ten cross-validation (*k*-fold cross-validation) method. The graph clearly shows that decision tree classification algorithm performs exceptionally well as compared to other algorithms taken into consideration.

Implementation of Classification Algorithm after CAE feature selection: Classification, grouping, and regression algorithms all utilise a training dataset to establish weight factors that may be applied to previously unseen data for predictive purposes. Prior to executing a data mining technique, it is required to narrow down the training dataset to the most relevant attributes. Dimensionality reduction is the process of modifying a dataset in order to extract only the characteristics required for training. Due to its simplicity and computational efficiency, dimension reduction is critical since it minimises overfitting. Thus, dimensionality reduction is critical throughout the data pre-processing phase. A correlation-based feature selection method selects

Table 3 Accuracy achieved by classification algorithm with CAE

Classification algorithm used for implementing CAE	Accuracy (%)	MAE	Precision	Recall
Naïve Bayes + CAE + *k*-fold cross-validation	87.51	0.058	0.878	0.875
Random forest + CAE + *k*-fold cross-validation	95.83	0.061	0.959	0.958
Decision tree + CAE + *k*-fold cross-validation	96.91	0.0169	0.969	0.969
Multilayer perceptron + CAE + *k*-fold cross-validation	97.68	0.0172	0.977	0.977
Decision Table + CAE + *k*-fold cross-validation	90.13	0.146	0.909	0.901
JRip + CAE + *k*-fold cross-validation	95.53	0.0252	0.956	0.955
Logistic regression + CAE + *k*-fold cross-validation	97.84	0.0087	0.979	0.978

attributes based on the usefulness of individual features for predicting the class label, as well as the degree of connection between them. We avoid strongly linked and irrelevant features. The correlation attribute evaluator determines an attribute's value in a dataset by calculating the correlation between the attribute and the class attribute. Nominal qualities are assessed individually, with each value acting as a signal. A weighted average is used to generate an overall correlation for a nominal characteristic. We picked the top ten attributes with a threshold value larger than 1 using the aforementioned attribute evaluator CAE in conjunction with the ranker search strategy.

The following table summarises the results of the implementation of several classification algorithms using CAE and the test option as *k*-fold cross-validation approaches. As shown in Table 3, the combination (logistic regression + CAE + *k*-fold cross-validation) achieved the greatest accuracy of 97.84% when compared to other classification algorithms such as naive Bayes, random forest, decision tree, multilayer perceptron, decision table and JRip. As can be seen, the multilayer perceptron technique improves accuracy to 97.68%, which is significantly higher than the accuracy obtained without utilising the feature selection approach. The remainder of the algorithms is also accurate to an acceptable level. Only the top fifteen attributes of the dataset are considered when implementing these methods. Other performance metrics such as mean absolute error (MAE), precision, and recall value are also considered in this table. As our dataset contains no outliers, we will use accuracy as our primary parameter for evaluating our classifier's effectiveness.

Figure 2 is a graphical illustration of the implementation of the classification algorithms discussed previously using CAE and cross-validation (*k*-fold cross-validation) as testing methods. The graph clearly demonstrates that the logistic regression algorithm outperforms all other algorithms considered. However, as illustrated in Fig 1, practically all classification systems obtain a prediction accuracy of greater than 90% (Fig. 4).

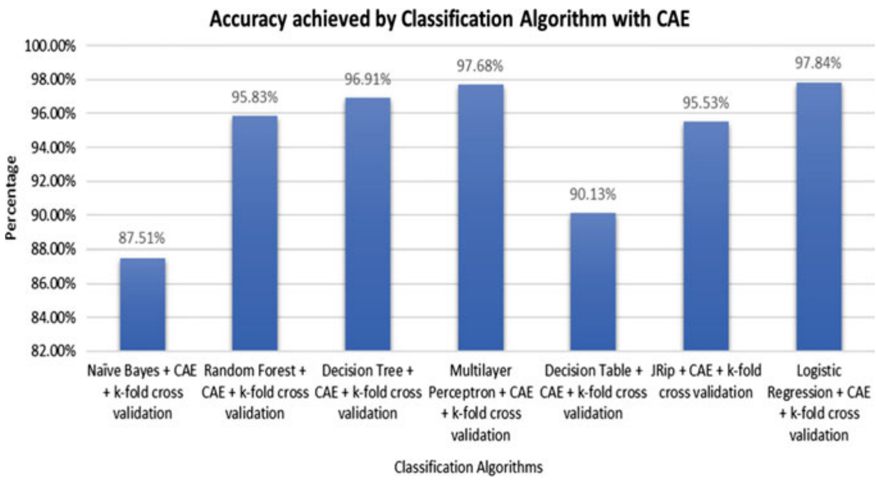


Fig. 4 Graphical representation of accuracy level of classification algorithms with CAE

Implementation of Bagging Ensemble after CAE feature selection: As part of this implementation, classification algorithms are applied to a dataset that has been reduced in features by employing the CAE feature selection technique in conjunction with the bagging ensemble and *k*-fold cross-validation selections, among other techniques. Table 3 shows that when compared to other classification algorithms taken into consideration, logistic regression and multilayer perceptron classification algorithms achieved the highest accuracy of up to 97.90%, as well as naive Bayes and random forest classification algorithms (also known as random forest and JRip classification algorithms). Using the multilayer perceptron technique, we can see that their prediction performance increased from 84.59% (without feature selection) to 97.90% (with feature selection). Using feature selection techniques, the performance prediction of the vast majority of algorithms improves significantly over time. A number of other performance metrics, including mean absolute error (MAE), precision, and recall, are taken into account in this table. We are just interested in accuracy in this example because the dataset does not contain any outliers; thus, we are only interested in accuracy when evaluating the performance of our classifier.

Figure 3 presents a graphical depiction of the data in Table 4, which is shown below the figure. When compared to other methods taken into consideration, the graph clearly demonstrates that logistic regression and the multilayer perceptron classification algorithm perform remarkably well. The accuracy of these two algorithms in terms of performance prediction is close to 97.90%, which is higher than the accuracy of the decision table method, which is also a rule-based classification system. Classification algorithms such as random forest, J48, decision table, and JRip attain accuracy levels of over 90% in several cases (Fig. 5).

Implementation of AdaBoostM1 Ensemble after CAE feature selection: As part of this particular portion of the implementation, classification algorithms are applied to

Table 4 Accuracy achieved by classification algorithm with CAE and bagging ensemble

Classification algorithm used for implementing bagging	Accuracy (%)	MAE	Precision	Recall
NB + CAE + bagging + *k*-fold cross-validation	89.67	0.0616	0.899	0.897
RF + CAE + bagging + *k*-fold cross-validation	94.60	0.0686	0.946	0.946
DT + CAE + bagging + *k*-fold cross-validation	95.99	0.0232	0.960	0.960
MLP + CAE + bagging + *k*-fold cross-validation	**97.90**	0.0297	0.968	0.968
DT (table) + CAE + Bagging + *k*-fold cross-validation	94.76	0.1501	0.950	0.948
JRip + CAE + bagging + *k*-fold cross-validation	96.45	0.0242	0.965	0.965
LR + CAE + bagging + *k*-fold cross-validation	**97.90**	0.0248	0.968	0.968

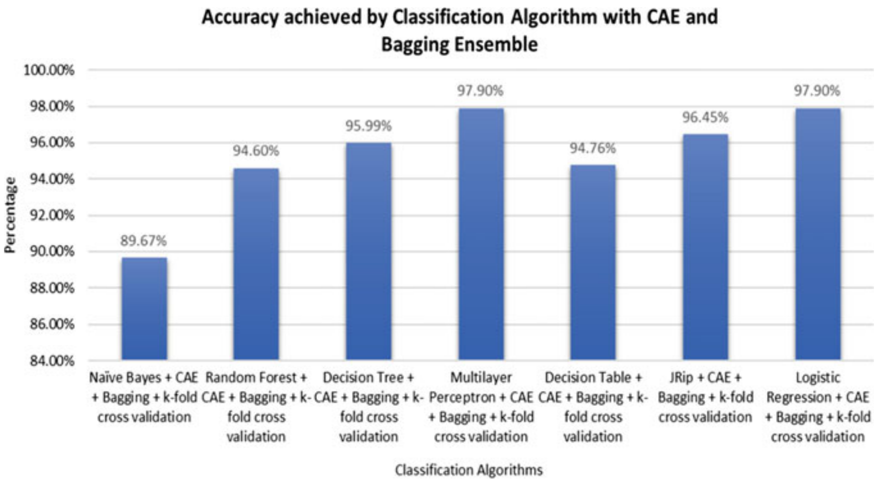


Fig. 5 Graphical representation of accuracy achieved by classification algorithm with CAE and bagging ensemble

a dataset that has been reduced in features by employing the CAE feature selection technique in conjunction with AdaBoostM1 ensemble learning with *k*-fold cross-validation, among other techniques. Table 4 shows that the decision table and logistic regression classification algorithms achieved the highest accuracy of up to 97.84% when compared to the other classification algorithms taken into consideration, which included naive Bayes, random forest, multilayer perceptron, decision tree, and JRip

Table 5 Accuracy achieved by classification algorithm with CAE and AdaBoostM1 ensemble

Classification algorithm used for implementing AdaBoostM1	Accuracy (%)	MAE	Precision	Recall
NB + CAE + AdaBoostM1 + *k*-fold cross-validation	91.83	0.0332	0.918	0.918
RF + CAE + AdaBoostM1 + *k*-fold cross-validation	95.83	0.0332	0.959	0.958
DT + CAE + AdaBoostM1 + *k*-fold cross-validation	97.38	0.0106	0.974	0.974
MLP + CAE + AdaBoostM1 + *k*-fold cross-validation	97.68	0.0131	0.977	0.977
DT(table) + CAE + AdaBoostM1 + *k*-fold cross-validation	97.84	0.010	0.979	0.978
JRip + CAE + AdaBoostM1 + *k*-fold cross-validation	97.07	0.012	0.971	0.971
Logistic regression + CAE + AdaBoostM1 + *k*-fold cross-validation	97.84	0.0087	0.979	0.978

classification algorithms. Using feature selection techniques, the performance prediction of the vast majority of algorithms improves significantly over time. A number of other performance metrics, including mean absolute error (MAE), precision, and recall, are taken into account in this table. One of the most interesting things about naive Bayes is that it achieves accuracy levels greater than 90%. We are just interested in accuracy in this example because the dataset does not contain any outliers; thus, we are only interested in accuracy when evaluating the performance of our classifier.

The graphical version of Table 5 is shown in the section Fig. 4. The graph clearly demonstrates that the decision table and logistic regression classification algorithms outperform all other algorithms taken into consideration when compared to one another. While the performance prediction accuracy of the decision table method is close to 97.84%, the accuracy of the JRip algorithm, which is also a rule-based classification system, is just slightly higher at 97.24%. Classification algorithms such as random forest, J48, decision table, and JRip attain accuracy levels of over 90% in several cases. However, the accuracy level of the naive Bayes algorithm has already risen to more than 90% (Fig. 6).

6 Analysis of All Predictive Classifier by MLHoEP Model

In this section, we will look at a comparative study of all of the algorithms that have been implemented. Using k-fold cross-validation as a testing option, we first examine the prediction accuracy of classification algorithms that use ensemble learning (bagging and AdaBoostM1) with and without ensemble learning. We can examine the following algorithms one by one in Table 6, which has a list of the algorithms taken into consideration:

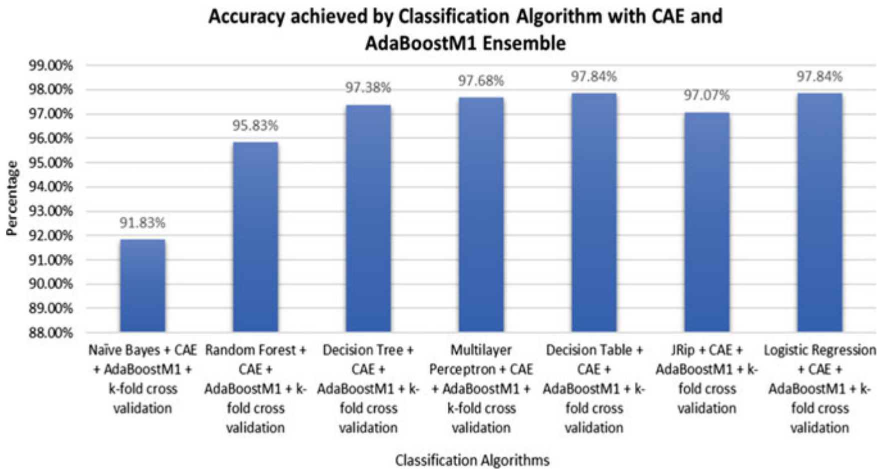


Fig. 6 Graphical representation of accuracy achieved by classification algorithm with CAE and AdaBoostM1 ensemble

Table 6 Accuracy achieved by all predictive classifiers by MLHoEP model

Classification algorithm	Accuracy ($P1$) (%)	Accuracy by CAE ($P2$) (%)	Accuracy by bagging ensemble ($P3$) (%)	Accuracy by AdaBoostM1 ensemble ($P4$) (%)
Naïve Bayes	86.59	87.51	89.67	***91.83***
Random forest	92.14	95.83	94.60	***95.83***
Decision tree	*96.76*	96.91%	95.99	**97.38**
Multilayer perceptron	84.59	97.68	***97.90***	97.68
Decision table	90.13	90.13	94.76	***97.84***
JRip	96.14	95.53	96.45	**97.07**
Logistic regression	87.51	97.84	**97.90**	97.84

Naïve Bayes: The ensemble learning methods used in our implementation included three different ensemble learning methods with k-fold cross-validation as the testing option method. Our observations revealed that the AdaBoostM1 ensemble performed exceptionally well on the supplied dataset, achieving the highest accuracy of 91.83%, which was significantly higher than that of another model like $P1$, $P2$, $P3$.

Random Forest: The ensemble learning methods used in our implementation included three different ensemble learning methods with k-fold cross-validation as the testing option method. Our observations revealed that the AdaBoostM1 ensemble performed exceptionally well on the supplied dataset, achieving the highest accuracy of 95.83%, which was significantly higher than that of another model like $P1$, $P2$, $P3$.

Decision Tree: The ensemble learning methods used in our implementation included three different ensemble learning methods with k-fold cross-validation as the testing option method. Our observations revealed that the AdaBoostM1 ensemble performed exceptionally well on the supplied dataset, achieving the highest accuracy of 97.38%, which was significantly higher than that of another model like $P1$, $P2$, $P3$.

Multilayer Perceptron: The ensemble learning methods used in our implementation included three different ensemble learning methods with k-fold cross-validation as the testing option method. Our observations revealed that the bagging ensemble performed exceptionally well on the supplied dataset, achieving the highest accuracy of 97.90%, which was significantly higher than that of another model like $P1$, $P2$, $P4$.

Decision Table: The ensemble learning methods used in our implementation included three different ensemble learning methods with k-fold cross-validation as the testing option method. Our observations revealed that the AdaBoostM1 ensemble performed exceptionally well on the supplied dataset, achieving the highest accuracy of 97.84%, which was significantly higher than that of another model like $P1$, $P2$, $P3$.

JRip: The ensemble learning methods used in our implementation included three different ensemble learning methods with k-fold cross-validation as the testing option method. Our observations revealed that the AdaBoostM1 ensemble performed exceptionally well on the supplied dataset, achieving the highest accuracy of 97.07%, which was significantly higher than that of another model like $P1$, $P2$, $P3$.

Logistic Regression: The ensemble learning methods used in our implementation included three different ensemble learning methods with k-fold cross-validation as the testing option method. Our observations revealed that the bagging ensemble performed exceptionally well on the supplied dataset, achieving the highest accuracy of 97.90%, which was significantly higher than that of another model like $P1$, $P2$, $P4$ (Fig. 7).

It is obvious from the preceding Fig. 5 that the AdaBoost1 ensemble approach did extraordinarily well in nearly all of the seven classification algorithms tested. It was discovered that classification algorithms such as naive Bayes and decision trees, as well as decision tables, random forests, and JRip, had higher accuracy than 97%. However, the bagging ensemble approach achieves the maximum accuracy in performance for multilayer perceptron and logistic regression, with a performance accuracy of up to 97.90%.

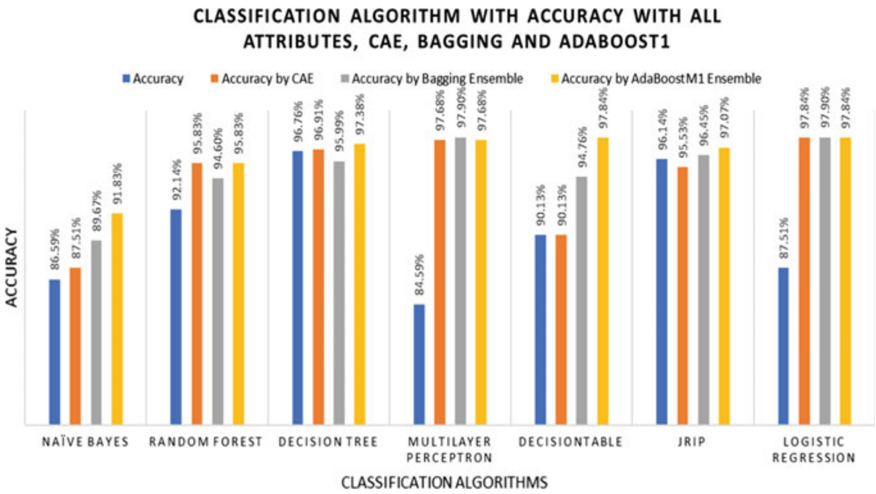


Fig. 7 Accuracy achieved by all predictive classifiers by MLHoEP model

7 Conclusion

When predicting the academic performance of students, many ensemble learning methods of data mining are taken into consideration. Feature selection methods such as bagging, boosting, and other ensemble learning methods are taken into consideration for implementation, as is the correlation attribute evaluator (CAE) as a feature selection algorithm. At the conclusion of this chapter, we can state that any classification algorithm that is implemented with the help of ensemble learning and the correlation attribute evaluator performs well when compared to algorithms that are implemented with ensemble learning but do not use the correlation attribute evaluator. It follows that ensemble learning, as well as feature selection, play an important role in improving the classification or prediction accuracy of the system.

Conflict of Interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

Acknowledgements This work has been carried out in the Department of Computer Science, Himachal Pradesh University, Summer Hill Shimla-5, India. This research facility has not been funded from the university or any other funding agency.

References

1. Abubakaria MS, Arifin F, Hungilo GG (2021) Redicting students' academic performance in educational data mining based on deep learning using TensorFlow
2. Ashraf M, Zaman M, Ahmed M (2018) Using predictive modeling system and ensemble method to ameliorate classification accuracy in EDM. Asian J Comput Sci Technol:44–47

3. Ayyappan G, SivaKumar K (2018) A noval approach of ensemble models by using EDM. Indian J Comput Sci Eng (IJCSE) 8:6
4. Bunkar K, Singh UK, Pandya B, Bunkar R (2012) Data mining: prediction for performance improvement of graduate students using classification. In: 2012 ninth international conference on wireless and optical communications networks (WOCN). IEEE, pp 1–5
5. Cano A, Zafra A, Ventura S (2013) An interpretable classification rule mining algorithm. Inf Sci:1–20
6. Christian TM, Ayub M (2014) Exploration of classification using NBTree for predicting students. In: International conference on data and software engineering (ICODSE), Nov, pp 1–6
7. Cortez P, Silva AM (2008) Using data mining to predict secondary school student performance
8. Gray G, McGuinness C, Owende P (2014) An application of classification models to predict learner progression in tertiary education. In: 2014 IEEE international advance computing conference (IACC), pp 549–554
9. Iam-On N, Boongoen T (2017) Improved student dropout prediction in Thai University using ensemble of mixed-type data clusterings. Int J Mach Learn Cybern 8:497–510
10. Jishan ST, Rashu RI, Haque N, Rahman RM (2015) Improving accuracy of students' final grade prediction model using optimal equal width binning and synthetic minority over-sampling technique. Decis Anal 2:1–25
11. Kalaivani S, Nalini S (2017) Analyzing student's academic performance based on data mining approach. Int J Innov Res Comput Sci Technol (IJIRCST):2347–5552
12. Katare A, Dubey S (2017) A comparative study of classification algorithms in EDM using 2 level classification for predicting student's performance. Int J Comput Appl 9:35–40
13. Kumar M, Singh AJ, Handa D (2017) Literature survey on educational dropout prediction. Int J Educ Manage Eng 2:8
14. Márquez-Vera C, Cano A, Romero C, Noaman AY, Mousa Fardoun H, Ventura S (2016) Early dropout prediction using data mining: a case study with high school students. Expert Syst 1:107–124
15. Mishra T, Kumar D, Gupta S (2014) Mining students' data for performance prediction. In: Fourth international conference on advanced computing & communication technologies, February, pp 255–262
16. Natek S, Zwilling M (2014) Student data mining solution–knowledge management system related to higher education institutions. Expert Syst Appl 14:6400–6407
17. Osmanbegovic E, Natek S, Zwilling M (2012) Data mining approach for predicting student performance. Econ Rev J Econ Bus 10:3–12
18. Osmanbegović E, Suljić M, Agić H (2014) Determining dominant factor for students performance prediction by using data mining classification algorithms. Tranzicija, pp 147–158
19. Tan M, Shao P (2015) Prediction of student dropout in e-Learning program through the use of machine learning method. Int J Emerg Technol Learn 10
20. Valsan V, Mathai PP, Babu I (2021) Monitoring driver's drowsiness status at night based on computer vision. In: 2021 international conference on computing, communication, and intelligent systems (ICCCIS), pp 989–993

Exposure of Sensitive Data Through Blockchain Wallets: A Comparative Analysis

Saba Khanum and Khurram Mustafa

Abstract Blockchain adoption has been at all-time high in last one decade. With widespread acceptance, blockchain is also subject to criticism from deployment to sensitive data exploitation in blockchain applications. Blockchain wallet is an essential part of blockchain framework and contains the most sensitive data of blockchain application. In 2020, out of 122 blockchain attacks, 27 were found to be solely focused on the wallet. Even though the comparison and analysis of various types of wallets are available on the market, there is no collective mapping of the best wallets in use. This study delves into wallet types, how they work and a comparative analysis of the state of the art on wallets that are currently in use. Presented research study is also helpful in designing a wallet and also gives assistance in choosing a wallet according to the requirement. The findings seem useful for developers and end-users at large, as it brings about the features of the current wallets in use and also proposes a novel design for blockchain wallets.

Keywords Blockchain Wallet · Vulnerablities · Attacks · Sensitive data protection

1 Introduction

According to the AIT news desk, the year 2020 was challenging from the standpoint of cybersecurity, as well as for the entire world in general. The global pandemic changes people's lifestyles and makes them more reliant on the digital world, providing cyber-attackers with a huge opportunity. Speaking of blockchain, according to the Atlas VPN team, hackers stole $43.78 billion in 122 attacks in 2020. There were 47 ethereum decentralised application attacks based on the ethereum smart contract, 28 breaches occurred through cryptocurrency exchanges, 27 attacks on blockchain

S. Khanum (✉)
Assistant Professor, MSIT, C-4 Janak Puri, New Delhi 110058, India
e-mail: saba@msit.in

K. Mustafa
Research Scholor, JMI, Jamia Nagar, Okhla 110025, India

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_13

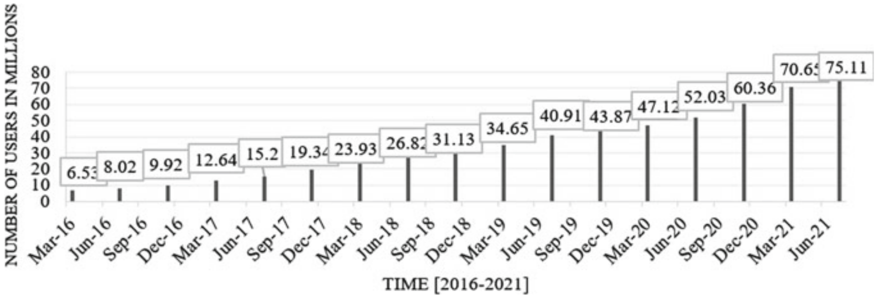


Fig. 1 Growth of blockchain wallet users worldwide

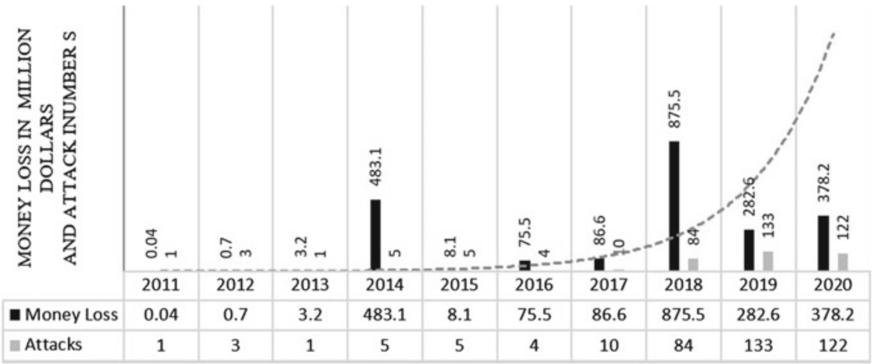


Fig. 2 Loss of cryptocurrency (m$) and attacks

wallets and 12 attacks on the blockchain itself [1]. Thus, the inherent nature and properties of blockchain wallets appear to prove more profitable and easily targetable for hackers. According to Statista, there will be 70 million cryptocurrency wallet users by March 2021 [2].The growth of the number of blockchain wallets users worldwide, as depicted in Fig. 1, reveals the tendency of acceptance.

With an escalation in acceptability of blockchain wallets, the number of attacks and money loss from wallets and other factors is also rising, as shown in Fig. 2.

Section 1 gives a brief introduction on why this research is imminent. Section 2 describes the typical wallets and their respective functioning. Section 3 illustrates the typical attacks occurrences on wallets and the comparative analysis. Section 4 proposes the novel design of the wallet. Section 5 briefly sheds light on challenges and future direction. Finally, Sect. 6 concludes with derived findings in proposed design of wallet, along with interpretation and perspective implications.

2 System Overview

Blockchain is a decentralised, distributed ledger used to record transactions and events. It had been introduced in 2009 by Nakamoto [3]. Blockchain revolution

Table 1 Comparison of blockchain wallet and traditionally used account

S. No.	Blockchain wallet	Traditional system
1.	One wallet to manage many cryptocurrencies	Needed separate account
2.	Easy exchange of funds	Time taking
3.	More secure	Less secure
4.	Easily accessible and help in maintaining privacy	Easily accessible
5.	Instant transactions across geographies without intermediates	Third-party is a pathway to transfer
6.	Low transaction fees	Comparatively large
7.	Easy currency conversion	Currency conversion is tedious

improves trust, transparency, and coherence in society by making the user more independent. It leverages a huge number of applied and successful applications, viz. agriculture field, land registry, e-voting supply chain, financial and power sector [4, 5], etc. To become part of the blockchain ecosystem, each user needs to possess a blockchain wallet. Blockchain wallets and e-wallet are digital wallets that help users to maintain, manage, and store assets or currency on their own. Both the wallets work in the same manner, but blockchain wallet uses blockchain technology for a transaction. Table 1 describes the comparative advantages of blockchain wallets over the traditional system.

The security of a wallet depends on how the user keeps the wallet. Exposure of data by any means like physical theft of the storage device, hacking and side-channel attack may leak the private key. Therefore, managing wallet and keys promisingly is very eminent. The types of wallet according to key management approaches are, viz. local key storage, online key storage wallet, offline key storage wallets and hosted wallets [7]. In local key storage, private keys are stored in the local storage of the device and remain accessible by software from the specified location. In online key storage, key management is done through the web browser. This wallet is generally known as a hot wallet. Hot wallets are user friendly. Hot wallets are available online all the time, viz. Coinbase and Blockchain.info. The private keys are stored on the cloud and hence can be accessed faster. Another type of wallet includes where wallets are password protected. The major drawback of this kind of key management system is that if the owner of the wallet forgets the password, then he/she loses the bitcoin balance of his/her bitcoin wallet [8]. In a paper wallet, keys are stored on paper in the form of bar codes. However, the drawback of this kind of storage is that the wallet is not immediately accessible. In brain wallets, a private key is memorised, and the transaction is done online. Secure key storage with hardware, viz. use of cold storage devices, most typically USB drives for keeping the private key and signing transactions. A critical analysis of a different variant of wallet is done through a risk analysis matrix based on crucial factors as given in Table 2 [11].

Through the risk analysis matrix, we become aware of the risks associated with the storage of wallets. The severity of risk is assessed as low, medium and high scale. Although storage of wallet plays a major role in the protection of wallet, network-level threats and attacks are still able to the deficit for the security of user wallet. The next section elaborates the attacks and vulnerabilities on wallets available in the market along with providing the detail of attacks held on a particular wallet.

3 Threat Analysis

This part of the research explores vulnerabilities and attacks in two categories of blockchain wallets, i.e. hot and cold. Hot wallets include the software wallet and hosted wallets as they are accessible to a user when a user is connected to a network. A cold wallet does the signing process of the transaction, and accessing private key offline rest is online. Hot wallets are maintained by the user itself or by third-party web services.

Table 2 shows the vulnerability of the wallet used currently in the market. It also indicates whether the wallet is hacked in the past. Both hot wallets and cold wallets available and currently in use are considered [10–15].

3.1 Comparative Analysis

For a wallet to use with ease and minimal protection in general, it must possess the following five features.

- *Control on private keys*—Wallets where you control your private keys.
- *Ease of use*—Elegant UI for ease of use.

Table 2 Risk analysis matrix of wallets

Category	Local key storage	Online storage		Offline key storage			Hosted wallet		
		Software wallets	Password protected wallet	Paper wallet	Brain wallet	Hardware wallet	Hot	Cold	Hybrid
Malware resistance	Yes-M	Yes-H	Yes-H	No	No	Yes-L	No	Yes-M	No
Resistant to physical theft	No	No	No	Yes-M	No	Yes-M	No	No	No
Resistant to physical observation	No	No	No	Yes-M	No	Yes-M	No	No	No
Resilient to password loss	Yes-H	Yes-H	Yes-H	Yes-H	Yes-H	Yes-M	No	No	No
Immediate access to funds	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Need of new user software	Yes	No	No	Yes	Yes	Yes	No	No	No
Cross-device portability	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

*Yes-H, M, L—high, medium, low risk of key getting compromised

Table 3 Vulnerabilities and attacks

	Vulnerability								Attacked	
Wallet name	Dusting attack	Phishing attack	Vulnerable signatures attack	Flawed key generation attack	Supply chain attack	Chip-level vulnerability	Bootloader vulnerability	Side-channel attack	Yes/no	The research team hacked the wallet
Ledger Nano X	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Ledger Nano S	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓
Trezor	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓
Atomic Wallet	✓	✓	✓	✓	✗	✗	✗	✗	✓	✓
Mycelium	✓	✓	✓	✓	✗	✗	✗	✗	✗	❍
Argent	✓	✓	✓	✓	✗	✗	✗	✗	✗	✓
Electrum	✓	✓	✓	✓	✗	✗	✗	✗	✓	✓
Jaxx	✓	✓	✓	✓	✗	✗	✗	✗	✓	✓
Exodus	✓	✓	✓	✓	✗	✗	✗	✗	✓	✓
My Ether wallet	✓	✓	✓	✓	✗	✗	✗	✗	✓	✓
Coinbase	✓	✓	✓	✓	✗	✗	✗	✗	✓	✓
Keepkey	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓

‘✓’ Not found

- *Development community*—Active development community.
- *Backup and security*—Backup and restore features.
- *Compatibility*—Compatible with different operating systems.

Table 3 shows the comparison of hot wallets, cold wallets and mobile wallets active presently in 2021. It is observed that hardware wallets are more secure but at the same time more costly compared to software wallets. The features like a smart contract, multi-signature, inheritance, vanity, multi-factor authentication and the requirement of individual identity data are significant from the security point of view. On the other hand, features like mobile support, desktop support, cloud storage, support for HD wallet and customer support facilitate the wallet.

4 Proposed Blockchain Wallet

After conducting a comparative analysis of the best available wallets in 2021, it was discovered that each feature included by the developer team is important and plays a significant role in the wallet's design. To achieve a wallet that is cost-effective, secure and needless personal info, we need a design of wallet which must-have characteristic as shown below with the help of fish bone diagram (Fig. 3; Table 4).

- *Hierarchical Determinism*: Use of new public key and private key for the signature of any new transaction.
- *Inheritance*: In case the owner of the wallet dies. What will be the future of crypto asset?
- *Attribute-Based Restriction*: There should be features like the daily limit of signing transactions of the money transfer and time-based signature.
- *Use of vault*: Vault feature helps user to split wallet and keep a wallet online having less information of asset and part of wallet offline for keeping valuable asset more secure.
- *Recovery*: In case the wallet is lost and no more accessible to the owner. How the owner can claim back?
- *Multi-Signature*: This feature is used to make anonymous transactions so that attackers can map transactions of a single user.

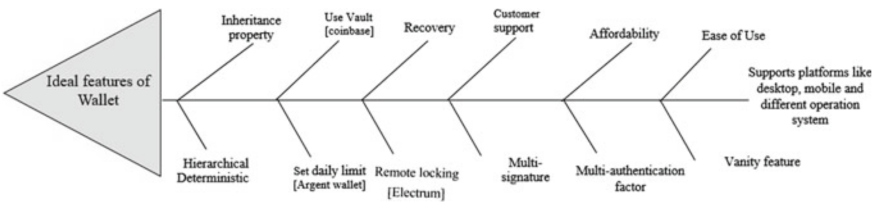


Fig. 3 Fish bone diagram of proposed blockchain wallet

Table 4 Comparison of existing wallets on vital parameters

Wallet name	Smart contract support	Multi-signature feature availability	Identity-based info. needed	Multiple OS support (Mac, Windows, Linux)	Mobile app support	Customer support	Wallet type	Native token	Inheritance property	Hardware wallet support	Backup advantage	Multifactor authentication	Cost effective	Multi-cryptocurrency	HD wallet	Cloud storage support	Vanity feature
Ledger Nano X	✓	✗	✓	✓	✓	✓	C	✗	✗	✓	✓	✓	✗	✓	✓	✗	✗
Ledger Nano S	✓	✗	✓	✓	✓	✓	C	✗	✗	✓	✓	✓	✗	✓	✓	✗	✗
Trezor	✓	✗	✓	✓	✗	✓	C	✗	✗	✓	✓	✓	✗	✓	✓	✗	✗
Atomic Wallet	✓	✗	✓	✓	✓	✓	C	✓	✗	✗	✓	✓	✓	✓	✓	✗	✗
Mycelium	✗	✗	✗	✗(MW)	✓	✓	H	✗	✗	✓	✓	✓	✓	✗	✓	✓	✗
Argent	✓	✓	✓	✗(MW)	✓	✓	H	✗	✗	✓	✓	✓	✓	✗	✗	✗	✓
Electrum	✗	✓	✗	✓	✓	✓	C	✗	✗	✓	✗	✓	✓	✗	✓	✗	✗
Jaxx	✗	✗	✗	✓	✓	✓	H	✗	✗	✗	✗	✗	✓	✓	✓	✓	✗
Exodus	✗	✗	✗	✓	✓	✓	H	✗	✗	✓	✓	✗	✓	✓	✗	✗	✗
My Ether wallet	✓	✗	✗	✓	✓	✓	H	✗	✗	✓	✗	✗	✓	✓	✗	✗	✗
Coinbase	✓	✓	✓	✓	✓	✓	H	✗	✗	✗	✓	✓	✓	✓	✗	✓	✗
Keepkey	✗	✗	✓	✓	✓	✓	C	✓	✗	✗	✓	✓	✗	✓	✓	✓	✗

* No information; *C* cold; *H* hot; *MW* mobile wallet

- *Customer Support*: In the present tech era, customer support is not just complimentary but an essential need that every customer.
- *Ease of Use*: Understandable and adaptable wallet got more popularity among the people.
- *Vanity*: It is a feature of the wallet which makes the public key and private key personalised.
- *Affordability*: We need a wallet that secures enough to afford enough.
- *Multifactor Authentication*: At least two-factor authentication on the wallet should be mandatory to provide by the wallet providers.

5 Challenges and Future Scope

The maintenance of wallet with private keys and sensitive information is an issue in current scenario. In order to build a wallet equip with features, viz. inheritance, vanity, multi-factor authentication and affordable, is itself a challenge. Blockchain technology is itself a revolution as there is no single data owner. Wallet is an integral part of this blockchain technology. With above-mentioned (Sect. 5) directions, amendment in the design of hot wallet and cold wallet is possible.

6 Conclusion

Blockchain wallet is part of the blockchain ecosystem which stores the most sensitive data of the user. Protecting the wallet aids the user identity, sensitive data security and improves belief in blockchain technology. The research article shows the features available and useful for the wallet. Research done in this article implies that features like inheritance, vanity, remote access, vault and the daily limit should be inbuilt in the wallet so that the sensitive asset will not get compromised or blocked. There is no such wallet available that fulfils all such criteria in a single wallet. Wallet being the imminent part of the ecosystem needs immediate attention.

References

1. Blockchain hackers stole $3.8 billion in 122 attacks in 2020. Available online https://aithority.com/technology/blockchain/blockchain-hackers-stole-3-8-billion-in-122-attacks-in-2020/. Accessed on 22 Jun 2021
2. Blockchain wallets 2011–2021. Statista. Available online https://www.statista.com/statistics/647374/worldwide-blockchain-wallet-users/. Accessed on 17 Aug 2021
3. Nakamoto S (2008) Bitcoin: a peer-to-peer electronic cash system. Decentralized Bus Rev 21260

4. Bao J, He D, Luo M, Choo K-KR (2020) A survey of blockchain applications in the energy sector. IEEE Syst J 15(3):3370–3381
5. Pournader M, Shi Y, Seuring S, Koh SCL (2019) Blockchain applications in supply chains, transport and logistics: a systematic review of the literature, vol 58, no 7, pp 2063–2081, Apr 2019. https://doi.org/10.1080/00207543.2019.1650976
6. Vishawjyoti (2021) E-Wallet, blockchain bus, pp 97–111, Feb 2021
7. Pal O, Alam B, Thakur V, Singh S (2021) Key management for blockchain technology. ICT Express 7(1):76–80
8. Furneaux N (2018) Analysis of recovered addresses and wallets, investig. Cryptocurrencies, pp 147–173
9. Aumasson JP, Shlomovits O (2020) Attacking threshold wallets. Cryptology eprint Archive
10. Cryptocurrency wallet hacks spark dustup. Threatpost. Available online https://threatpost.com/cryptocurrency-wallet-hacks-spark-dustup/140445/. Accessed on 04 Aug 2021
11. Cold and hot wallets are vulnerable to attacks! GK8—you can't hack what you can't reach. Available online https://www.gk8.io/cold-and-hot-wallets-are-vulnerable-toattacks/. Accessed on 29 Jun 2021
12. Cox D (2021) Best crypto wallet for 2021, top 10 crypto wallets. Available online https://www.cryptonewsz.com/crypto-wallet/. Accessed on 31 Aug 2021
13. Mcnamara R (2021) Best cryptocurrency wallets—top 5 wallets for BTC—Benzinga, 2021. Available online https://www.benzinga.com/money/best-crypto-wallet/. Accessed on 31 Aug 2021
14. Lawrence (2021) Overview of the best crypto wallets 2021, Akinpedia. Available online https://www.akinpedia.com/overview-of-the-best-crypto-wallets-2021/. Accessed on 31 Aug 2021
15. Martindale J (2021) The best crypto wallets for storing bitcoin, ethereum and dogecoin. Available online https://www.forbes.com/sites/forbes-personal-shopper/2021/07/19/bestcrypto-wallet/?sh=657fd0662b00. Accessed on 31 Aug 2021

Classification of Sentiment Reviews for Indian Railways Using Machine Learning Methods

Manju Bagga, Ritu Aggarwa, and Nitika Arora

Abstract AI provides the concept of machine learning that helps to automate the decision-making process by analyzing data inputs. It trains machines by providing it sample data and thus makes the system intelligent that is helpful for real-world AI applications. Machine learning algorithms are applied to such social feedback data to excerpt useful information that confers a competitive edge to several enterprises. There are enough machine learning technologies in the existing literature on sentiment analysis. However, it still needs optimizations for a better decision making process for several enterprises. In this paper, we proposed a scheme for Indian Railways for determining sentiments from Facebook. This is a more specific scheme that clouts business intelligence over different classifiers, viz. SVM, NB, RF, and decision tree, K-NN. The proposed scheme is provided with various parameters like F-measure, recall, precision, logarithmic loss, and accuracy. The first section of this paper provides the preface of sentiment analysis, and the next section presents the related work and motivation for sentiment analysis then methodology adopted for better decision making through machine learning to bring out in depth knowledge for future marketing game plans; it then discussed the experimental results, and finally, the paper encapsulates the conclusion and future scope in the area of sentiment analysis.

Keywords Sentiment analysis · Machine learning · DT · NB · SVM · RF · Logarithmic loss

M. Bagga (✉) · R. Aggarwa · N. Arora
MMICT & BM, MMDU, Mullana, Ambala, Haryana, India
e-mail: rathi.ancester@gmail.com

N. Arora
e-mail: nitika.arora@mmumullana.org

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_14

1 Introduction

In past years, the different kinds of sharing applications and opinions are rising in the different fields. Different social sites like Facebook, Twitter provide to access the users to post their comments and reviews regarding any topics issues and projects. With the help of these, we can see the reviews that are shared by the other peoples and make importance for us. Many commercial enterprises like business firms, airways, railways, Twitter, Facebook, Instagram can make use of sentiment analysis so that the satisfaction level of user can be examined in the direction of services, issues, and products. Sentiment analysis accustomed in accord with user's attitude. Sentiment analysis revolves around making decisions that requires adequate information to acquire and refine it that investigations could be available by different applications of AI. Well-established and standardized process, i.e., machine learning, is really a great language for making decisions in established areas.

Sentiments can be defined as a view and opinion someone that expressed or being held, and the analysis means when people review it accordingly their opinions. This framework provides the training and testing phase to guide the different operations related to it. The methodology behind this SA used the evaluation method with the following metrics like accuracy, precision, recall, F-measure, and logarithmic loss are that based on some matrix that is confusion matrix with TruePositive (T_Pos), False-Positive (F_Pos), TrueNegative (T_Neg), and FalseNegative (F_Neg). In this paper, we examine the efficiency of supervised learning techniques, namely the support vector machine, naïve Bayes, random forest, and decision tree, K-NN for sentiment analysis of Indian Railways from Facebook. K-NN is capable of performing classification and regression, effectual for high dimensional spaces, and mostly used in data mining experimentation.

2 Related Work

This briefly surveys related work on sentiment classification techniques using machine learning [1]. We analyzed Twitter sentiment classification using machine learning techniques. Recurrent neural network (RNN) is studied in [2] for document-level sentiment analysis [3]. We opposed the traditional neural network word embedding and proposed new sentiment-specific word embedding for sentiment classification [4]. We considered the n-gram machine learning method used for sentiment analysis. Other related work on text classification using deep learning methods is suggested in [5]. A novel method is proposed in [6] for merging lexical- and learning-based techniques for sentiment classification. An approach to sentiment analysis using lifelong (LL) method is proposed in [7]. Daeli et al. [8] discussed the sentiment analysis of people for movie reviews by using K-nearest neighbor and information gain to bring a successful conclusion [9]. It studied the Twitter and Reddit posts for real-time prediction for the prices of bitcoin using sentiment analysis; various algorithms of machine learning are analyzed for variations in bitcoin price. In [10], for sentiment analysis, it evaluated different machine learning algorithms for Czech language. For categorization of ideas, a material analysis way is used.

3 Methodologies for Sentiment Analysis

The methodology is used for the Facebook accounts with the dataset collection by using the sentiment classification for Indian railways. The Facebook API in which we have the live connectivity and reviews is based on the classification technique in which the preprocessed datasets are used for training and testing. The following subsections provide detailed methodologies that are to be employed.

3.1 Problem Formulation

Different social networking sites like Twitter, Facebook, Instagram a few of these mention how the information changed and dispersed or shared frequently. These are the platform that is used for social gathering, meetings, and chats or for information exchange and transfer. These platforms provide the better opportunity to connect with the people and deliver the valuable feedback about on products and their services.

3.2 Framework for Proposed Methodology

A framework is employed for this proposed work that used the sentiment classification techniques used for guiding the research work. The sentiments are the opinions and reviews on social media, and comments used on social media sites of Indian railways are collected from the Facebook Web site, and these sentiments are the reviews or the knowledge about the people sentiments. In this framework, the different machine learning algorithms and approaches are used for sentiment analysis like naïve Bayes, SVM, K-NN, random forest, decision tree, etc.

Figure 1 shows the framework. In the sentiment, classifications are two-phase training and testing. In the training phase to train the datasets for preprocessing and the train, datasets are tested for analysis of the system. The results are based on some classification metrics of their accuracy performance like positives, negative, and mixed-mode; there is a need to know about each algorithm that accurately finds the classification necessity.

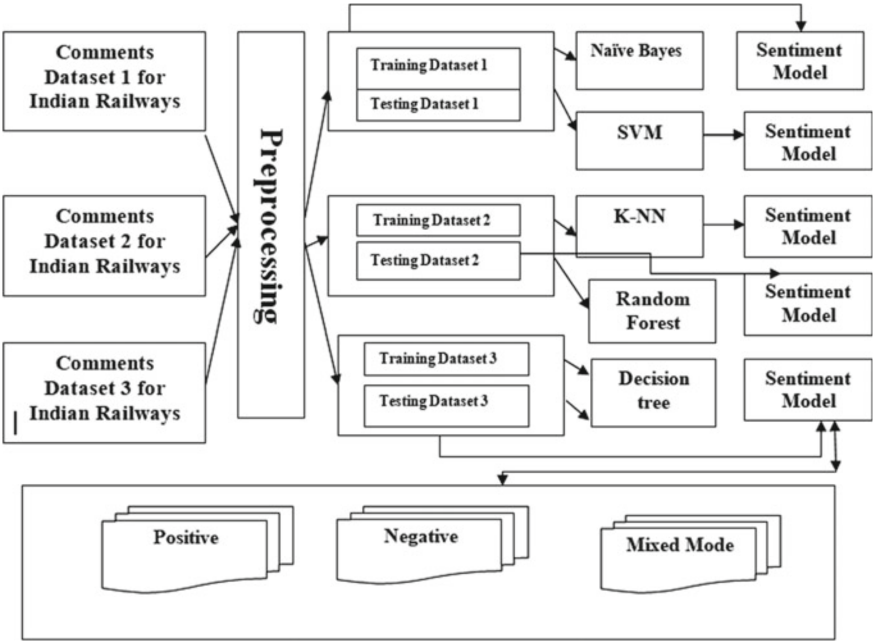


Fig. 1 Framework for sentiment classification

3.3 Evaluation Method

In this proposed work, the methodology is used for sentiment analysis evaluation with the help of machine learning techniques and their algorithms which extract and collect the utility of information by the various comments and reviews from the Indian Railways.

Figure 2 shows the evaluation metrics. Table 1 describes the confusion matrix which describes the results for according to their prediction values based on the classification metrics. Prediction of classification accuracy is as follows: the correct/incorrect prediction results are TruePositive (T_pos), FalsePositive (F_Pos), FalseNegative (F_Neg), TrueNegative (T_Neg). Calculation of metrics is shown in Table 1.

4 Experimental Results

Experiments are calculated by using ML approaches. SVM and K-NN datasets are used against the Indian Railways collected from the Facebook accounts. The observed F_Pos rate is mentioned below according to each technique. As shown in Table 2, the FPR is presented for all machine learning algorithms.

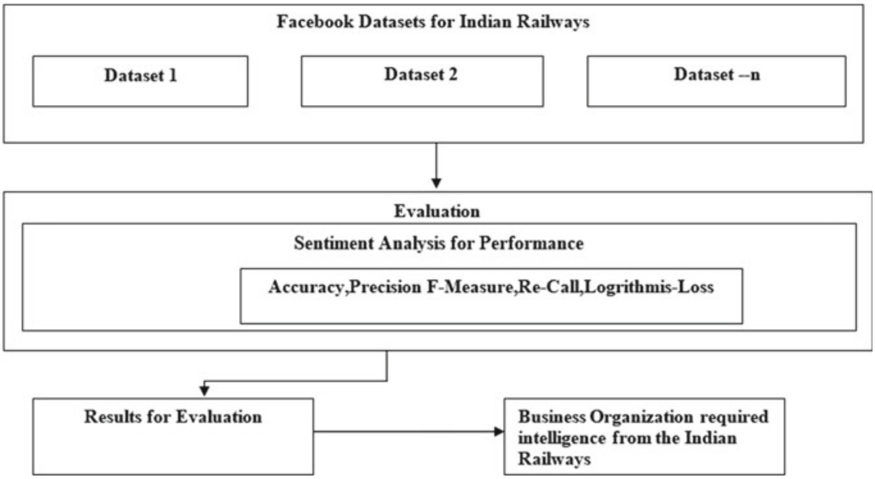


Fig. 2 Procedure for evaluation

Table 1 Metrics calculation

Accuracy	= T_Pos + T_Neg/T_Pos + T_Neg + F_Pos = F_Neg
Precision	= T_Pos/T_Pos + F_Pos
Recall	= T_Pos/T_Pos + F_Neg
F-measure	= 2 * ((precision * Recall)/(precision + Recall)
Log loss	$= -\frac{1}{N} = \sum_{i=1}^{n} (Yi \log pi + (1 - Yi)\log(1 - pi))$

Table 2 Results for false positives

Machine learning algorithm	F_Pos rate
Decision tree	8.8
K-NN	6.1
Naïve Bayes	9.4
SVM	7.1
Random forest	7.9

It is shown that K-NN has better performance with the least percentage of false positives. Naive Bayes is the algorithm that showed the highest number of false positives. Higher FPR indicates less in performance in terms of false positives. The computed accuracy, precision, recall, and *F*-measure, logarithmic loss are presented against the machine learning techniques in Table 3.

In Fig. 3, *y*-axis represents the supervised machine learning experiments. The *x*-axis shows the performance by using classification accuracy metrics In this model, the K-NN, SVM, and RF showed high precision and recall classification for SA so

Table 3 Performance using machine learning approaches for sentiment analysis

Algorithms	Accuracy	Precision	F-measure	Recall	Logarithmic loss
Decision tree	88.5	83.2	82	82.8	79.8
Naive Bayes	87	82.3	80.5	79	74
SVM	90	85.5	86.5	84	80.2
Random forest	89.5	88.2	86	83.5	80.5
KNN	90.2	86	87	84.3	80

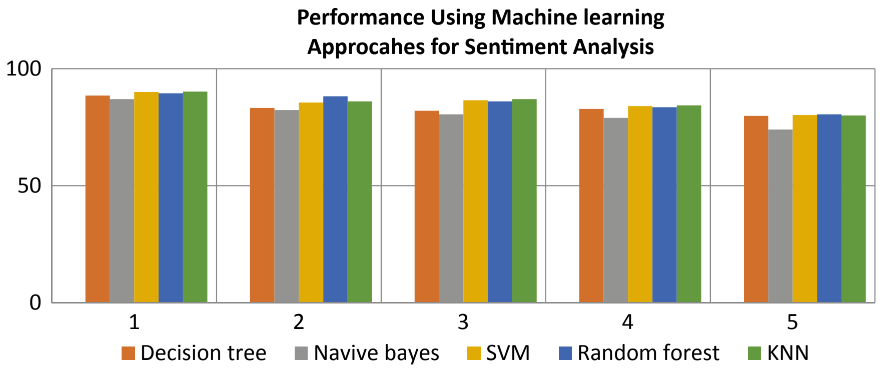


Fig. 3 Performance using machine learning approaches for sentiment analysis

that the K-NN has the highest F-measure value concerning other algorithms. K-NN shows a better performance for SA.

5 Conclusion and Future Scope

In this paper, we presented a methodology for sentiment classification of Indian Railways from Facebook; it has two stages, training and testing. In the training stage, different models can be designed through classifiers. In the testing stage, these models are passed for labeling the unlabeled comments online. The results show that K-NN outperforms when compared with the decision tree, NB, and RF in the presented methodology. There exist some possible extensions to our work for a wide variety of annotations to simplify sentiment classification and retrieval. Also, the presented methodology is flexible enough so that it is applicable in the developing areas of sentiment classification.

References

1. Madhuri DK (2019) A machine learning based framework for sentiment classification: indian railways case study. Int J Innov Technol Explore 8(4):441–445
2. Tang et al (2015) DocumentModeling with gated recurrent neural network for sentiment classification. In: Proceedings of the 2015 conference on EMNLP, pp 1422–1432
3. Tang et al (2014) LearningSentimentSpecific word embedding for twitter sentiment classification. In: Proceedings of the 52nd annual meeting of the AC linguistics, vol 1: long papers, pp 1555–1565
4. Tripathy A, Agrawal A, Rath SK (2016) Classification of sentiment reviews using the n-gram machine learning approach. Expert Syst Appl 57:117–126
5. Zhang X, Zhao J, LeCun Y (2015) Character_level convolutional networks for text classification. In: Advances in neural information processing systems, pp 649–657
6. Zhang L, Ghosh R, Dekhil M, Hsu M, Liu B (2011) Combining lexicon based and learning based methods for Twitter sentiment analysis. Tech Rep HPL 2011:89
7. Chen Z, Ma N, Liu B (2018) Lifelong learning for sentiment classification. arXiv:1801.02808
8. Daeli NOF, Adiwijaya A (2020) Sentiment analysis on movie reviews using information gain and K-nearest neighbor. J Data Sci Appl 3(1):1–7
9. Raju SM, Tarif AM (2020) Real-time prediction of BITCOIN price using machine learning techniques and public sentiment analysis
10. Arote Rutuja S, Gaikwad Ruchika P, Late Samidha S, Gadekar GB (2020) Online shopping with sentimental analysis for furniture shop. IRJMETC Int Res J Modern Eng Technol Sci 02(05):1–8

A Review on Community Detection Using Deep Neural Networks with Enhanced Learning

Ranjana Sikarwar, Shashank Sheshar Singh, and Harish Kumar Shakya

Abstract Community detection has become pervasive in understanding complex network structures and detecting similar patterns. The main motivation behind using deep learning methods for community detection comes from the brilliant performance results shown by deep neural networks in various fields. Using unsupervised learning models, the problem of community detection can be solved. The high-dimensional feature space representation of the network data leads to a complex neural network architecture that requires a high number of trainable parameters. Deep learning-based models can transform the high-dimensional graph data of complex networks into simple, low-dimensional space or latent representation. The transformation of network representation to latent representation consists of meaningful features of the network data. This mapping preserves the structural information of the network later on, which clustering algorithms can be applied to the converted latent representation. This survey paper provides an overview of the traditional and deep learning-based methods of community detection, followed by a discussion on the challenges and future directions of community detection.

Keywords Community detection · Deep learning · Graph partitioning · Spectral clustering · Statistical inference

R. Sikarwar (✉) · S. S. Singh · H. K. Shakya
Department of Computer Science and Engineering, Amity University, Gwalior, Madhya Pradesh 474005, India
e-mail: ranjana.sik@gmail.com

S. S. Singh
e-mail: shashank.sheshar@thapar.edu

H. K. Shakya
e-mail: hkshakya@gwa.amity.edu

Thapar Institute of Engineering and Technology, Patiala, Punjab 147004, India

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_15

1 Introduction

Community detection is a multidisciplinary research area that is used to study the structural properties of complex networks. These structures of the network have a high-dimensional form of graph data, which requires a large number of trainable parameters. Deep learning techniques are employed to analyze the rich nonlinear structure of real-world networks. Spectral clustering techniques cannot scale for large networks because they can perform well only for small networks. Other traditional methods (as mentioned in Table 1) used for community detection, such as statistical inference, do not perform well on large networks having high-dimensional features. Such networks have high computational complexity in terms of both time and space. Deep learning models when used for community detection in networks provide improved performance over traditional techniques like spectral clustering and statistical interference.

They learn the nonlinear structural properties of the network and represent the network in its low-dimensional form. Reference [7] presented the architecture of the convolutional neural network to mitigate the redundant information existing in the

Table 1 Taxonomy of traditional community detection methods

Traditional approaches	Summary	Advantages	Disadvantages
Top-down [1, 2]	The whole network is represented as a graph and divided into communities slowly	Able to detect overlapping communities easily	Large processing delays when overlapping of communities becomes high
Bottom-up [3, 4]	Local structures are taken into consideration and, later on, expanded into communities in the overall network	Executes in linear time mostly	Fail to detect small communities many times because local structures at the beginning do not observe these small communities and expansion strategies do not include the nodes belonging to these small communities
Data structure-based [5, 6]	The network is converted into data structures (often in tree form) for further processing of community detection	Efficient community detection using data structures like tree, threaded binary tree, or spanning-tree when combined with other approaches like parallel processing techniques	Very high computational costs for converting an entire network of millions of nodes and edges into data structure form

networks by sharing the weights of convolutional layers among residual blocks, thus showing a 45% reduction in network parameters and increased efficiency.

1.1 Community: Definition and Properties

Community: A set of communities $c = c_1, c_2..., c_k$ denotes k communities partitioned in a network $G\,(V, E)$. Informally, a community C is a subgraph of a network that consists of a collection of nodes V such that the number of edges inside the community is denser than the edges linking the vertices of C with other communities of the graph. Here, V is a set of vertices; E is set of edges, $n = |V|$, $m = |E|$, $C = A$ subset of V, $n_c = |C|$.

Intra Cluster density

$$\delta_{\text{in}}(C) = \frac{\#\text{internal edges of } C}{n_c(n_c - 1)/2}$$

Inter Cluster density

$$\delta_{\text{ext}}(C) = \frac{\#\text{inter - cluster edges of } C}{n_c(n - n_c)}$$

$$\partial_{\text{ext}}(C) \ll 2m/n(n-1) \ll \partial_{\text{in}}(C)$$

Connectedness is an important property that maintains the connections between each pair of vertices in C. Community detection is delineated on sparse graphs only.

2 Categorization of Deep Learning Techniques

A. **Convolutional Neural Networks (CNNs)**

It belongs to a specific category of feed-forward neural networks. Reference [8] addresses the problem of community detection in topologically incomplete networks. Their work proposed a deep CNN model that showed more robustness than classical supervised models, even in the case of missing edges in networks.

B. **Autoencoder-Based CD Approach**

Autoencoders (AEs) are unsupervised models similar to spectral clustering frameworks using low-dimensional matrix reconstruction [9]. Several studies have been proposed to use variants of autoencoder models such as stacked autoencoder [10] and sparse AEs [11].

C. Generative Adversarial Networks (GANs)

GANs are comprised of two competing neural networks with adversarial training to improve the discriminative ability when applied to community detection problem solves the overfitting challenge and resulting in fast-adjusting precision. Wang [12] introduced a low-dimensional vector space graph representation approach. Here, each vertex of the graph is represented as a low-dimensional vector space. A novel deep learning algorithm was proposed by the authors [13] to utilize graph representation learning techniques to solve overlapping community detection problems. Previous approaches focused only on communities having domain-specific rich topological information and failed in the networks having less structural information. An approach for cross-domain network representation was devised by the authors in [14].

D. Deep NMF-Based CD Approaches

Non-negative factorization (NMF) [15] computation involves the factorization of a large matrix into two matrices having non-negative values. NMF approach follows for community detection tasks by decomposing the adjacency matrix of a network into the product of two matrices with non-negative elements. The error function is also minimized for further network partitioning tasks. NMF can be implemented in both overlapping and non-overlapping community detection tasks. Conventional NMF cannot capture all the sophisticated topological information for community detection. The deep learning-based NMF approach proposed for the multilayer learning strategy of complex data to uncover latent feature hierarchies using stacked NMF has shown an improved performance compared to that used for single-layered networks [16].

E. Deep SF-Based CD Approach

Sparse filtering (SF) [17] is an effective feature learning algorithm that is known to handle high-dimensional graph data. It is an efficient two-layer learning model which is hyperparameter-free with only a single hyperparameter. It optimizes the cost function-sparsity of l_2-normalized features and can scale easily to high-dimensional input data. Also, it is capable of learning significant features in multiple layers using stacked layering. In the discovery of communities, a sparse filtering algorithm is applied to extract the network features for further network partitioning tasks, resulting in meaningful community structures [18].

F. Community Embedding-Based Approaches

The graph embedding approach focuses on the distribution of nodes present in communities in low-dimensional space. The approach embeds communities rather than specific nodes, which is a reverse approach. Community embedding is good for community detection as well as node categorization [19]. Reference [20] proposed a probabilistic generative model to learn representations of the social network by observing the information diffusion cascades instead of network structures. The proposed model learns community-preserving social network embeddings from social contagion logs. The focus is to discover social structures and predict information propagation in the network.

G. **Community Detection Based on Graph Neural Networks (GNN)**

GNNs can model the complex relationships in graph-related data. GNN models are based on deep learning and graph mining techniques. The authors have proposed a model for detecting overlapping communities using graph neural networks (GNN) approach. This GNN-based model has proved to be more effective and robust than other existing approaches [21]. The authors have proposed a modified GNN framework that involves a line graph and a non-backtracking operator of the graph to analyze edge adjacency knowledge. The algorithms can be used for node-classification challenges apart from community detection tasks and have shown improvements in supervised community discovery problems [22].

2.1 Discussion

Table 2 summarizes the comparative analysis done based on the comprehensive survey of using deep neural networks (DNNs) for community detection. DNN is a new approach for solving social network analysis problems and is highly effective in graph and node representation in a network [23]. Previous approaches like stochastic block models and modularity maximization provide linear mapping to low-dimensional space. But mostly real-world networks have nonlinear structures so deep neural networks have proved to be effective in nonlinear representation [24].

3 Challenges and Future Directions

The models and frameworks we have discussed above are the most recent strategies developed in the last few years to solve the community detection problem. From our study, we have encountered many challenging criteria that need to be further focused on, and the most efficient solutions must be provided. This section presents some of these challenges spotted in our study that may lead to a new future research direction.

Temporal Changes in Communities

As the network continuously reflects changes in user relations and topological information, dynamic community detection should be considered. The models with high computational power which can analyze the dynamic community detection and extract spatio-temporal characteristics of the social structures are in demand to be developed.

Table 2 Comparative study of deep learning techniques on community detection

References	Algorithm/methodology used	Advantages
Perozzi et al. [25]	Unsupervised feature learning technique called DEEPWALK is used. Perform random walks at graph vertices to learn representations including neighborhood information, community membership, and vertex resemblance	Flexible to reach out to all parts of the network, model is adaptable to even small changes without overall computation
Tang et al. [26]	Scalable to very large networks: directed, undirected, or weighted networks, preserves both local and global network structures	Improvement over classical stochastic gradient descent using edge-sampling algorithm
Grover et al. [27]	An improved approach over [25] called Node2vec using biased random walk to learn topological structure	Scalable and much richer representation is achieved by making random walks biased resulting into exploring diverse neighborhoods provides better representation
Tran [28]	Autoencoder architecture is trained to simultaneously achieve link prediction and node classification	Can learn latent features for nodes in sparse, bipartite graphs with directed/weighted edges, parameter sharing reduces computationally complexity
Jia et al. [13]	CommunityGAN a framework for learning better graph representations	Solves the problem of overlapping community detection
Dhilber et al. [24]	Deep neural network with stacked autoencoders	Implementing nonlinear structures of real-world networks which overcomes the limitation of stochastic block models and modularity maximization
Li et al. [29]	WCD algorithm based on deep sparse autoencoder	More accurate community structures are detected as compared to the one found by using k-means algorithm implementing high-dimensional adjacency matrix directly

Meaningful Representation of Datasets

Generally, an enormous amount of data is generated by social networks, which is used as input datasets to predict communities. Deep learning techniques must use datasets in a meaningful format to predict the correct semantic representation of communities. Additionally, a better interpretation of different communities formed may help in fast information propagation.

No Prior Knowledge of the Number of Communities

According to [30, 31], random walks were performed to get preliminary communities and refine results by modularity. But in the case of disconnected networks, random walks cannot cover each node, thus degrading the performance of community detection algorithms.

Signed Networks

The impact of the type of relationship (positive or negative) on nodes is different. So, existing community detection approaches implemented on unsigned networks cannot be used for signed networks. So, to detect communities in signed networks the focus of the research has to be on representing negative ties. Deep learning strategies developed should be efficient enough to represent positive and negative ties in signed networks. Future work may cover the impact of signed edges.

Community Overlapping Detection

More efforts are needed to focus on the overlapping detection approaches as some of these strategies discussed in this survey have worked on overlapping community detection problems.

Efficient use of Computational Resources

Some of the developed algorithms require heavy computations, one such mentioned in where the processing of adjacency matrix to similarity matrix construction requires large computation resources. Thus, mechanisms for better use of computational resources should be developed for the new computation-specific strategies.

Comparative Analysis Intermediaries

There is a shortfall of straightforward comparative analysis techniques for the strategies we have studied so far. In this regard, the Network kit is the most widely used tool kit for large-scale network analysis tasks and has inbuilt algorithms already implemented by the researchers.

NLP Embeddings

The latest trend used is called random walks for node embeddings. These node embeddings help similar nodes remain close in their representations. So, further trends also include temporal graphs and ego networks.

4 Conclusion

In this survey paper, we analyzed the existing community detection techniques and current trends using deep learning approaches for community discovery tasks in various scenarios. As discussed in this review paper, deep learning models for community detection have emerged to be more robust, effective, efficient, and flexible to handle high-dimensional network data. However, there is a scope for more research work in future to be done on studying the overlapping community detection problem, need of optimized algorithms with less computational complexity, taking into account signed networks, meaning representation of datasets to predict correct number of communities, dynamic community detection as the network is undergoing changes continuously, etc. Finally, along with the taxonomy of traditional and deep learning methods, challenges and prospects for community detection have also been elaborated in this paper.

References

1. Qiao S et al (2018) A fast parallel community discovery model on complex networks through approximate optimization. IEEE Trans Knowl Data Eng 30(9):1638–1651. https://doi.org/10.1109/TKDE.2018.2803818
2. Lu Z, Sun X, Wen Y, Cao G, La Porta T (2015) Algorithms and applications for community detection in weighted networks. IEEE Trans Parallel Distrib Syst 26(11):2916–2926. https://doi.org/10.1109/TPDS.2014.2370031
3. Džamić D, Aloise D, Mladenović N (2019) Ascent–descent variable neighborhood decomposition search for community detection by modularity maximization. Ann Oper Res 272(1–2):273–287. https://doi.org/10.1007/s10479-017-2553-9
4. Pirouz M, Zhan J (2018) Optimized label propagation community detection on big data networks. ACM Int Conf Proc Ser:57–62. https://doi.org/10.1145/3206157.3206167
5. Souravlas S, Sifaleras A, Katsavounis S (2019) A parallel algorithm for community detection in social networks, based on path analysis and threaded binary trees. IEEE Access 7:20499–20519. https://doi.org/10.1109/ACCESS.2019.2897783
6. Souravlas S, Sifaleras A, Katsavounis S (2020) Hybrid CPU-GPU community detection in weighted networks. IEEE Access 8:57527–57551. https://doi.org/10.1109/ACCESS.2020.2982227
7. Boulch A (2018) Reducing parameter number in residual networks by sharing weights. Pattern Recognit Lett 103:53–59. https://doi.org/10.1016/j.patrec.2018.01.006
8. Xin X, Wang C, Ying X, Wang B (2017) Deep community detection in topologically incomplete networks. Phys A Stat Mech Appl 469:342–352. https://doi.org/10.1016/j.physa.2016.11.029
9. Cao J, Jin D, Yang L, Dang J (2018) Incorporating network structure with node contents for community detection on large networks using deep learning. Neurocomputing 297:71–81. https://doi.org/10.1016/j.neucom.2018.01.065
10. Liang Y, Cao X, He D, Chuan W, Xiao W, Weixiong Z (2016) Modularity based community detection with deep learning. IJCAI Int J Conf Artif Intell 2016:2252–2258
11. Tian F, Gao B, Cui Q, Chen E, Liu TY (2014) Learning deep representations for graph clustering. Proc Natl Conf Artif Intell 2:1293–1299
12. Wang H et al (2021) Learning graph representation with generative adversarial nets. IEEE Trans Knowl Data Eng 33(8):3090–3103. https://doi.org/10.1109/TKDE.2019.2961882
13. Jia Y, Zhang Q, Zhang W, Wang X (2019) CommunityGan: Community detection with generative adversarial nets. In: Web conference on 2019—proceedings of world wide web conference on WWW 2019, pp 784–794. https://doi.org/10.1145/3308558.3313564
14. Xue S, Lu J, Zhang G (2019) Cross-domain network representations. Pattern Recognit 94:135–148. https://doi.org/10.1016/j.patcog.2019.05.009
15. Lee DD. Learning the pars of objects by nonnegative matrix factorization
16. Song HA, Kim BK, Xuan TL, Lee SY (2015) Hierarchical feature extraction by multi-layer non-negative matrix factorization network for classification task. Neurocomputing 165:63–74. https://doi.org/10.1016/j.neucom.2014.08.095
17. Ngiam J, Koh PW, Chen Z, Bhaskar S, Ng AY (2011) Sparse filtering. In: Advance neural information processing system 24, 25th annual conference on neural information processing system, NIPS, pp 1–9
18. Xie Y, Gong M, Wang S, Yu B (2018) Community discovery in networks with deep sparse filtering. Pattern Recognit 81:50–59. https://doi.org/10.1016/j.patcog.2018.03.026

19. Cavallari S, Zheng VW, Cai H, Chang KCC, Cambria E (2017) Learning community embedding with community detection and node embedding on graphs. Int Conf Inf Knowl Manage Proc Part F1318:377–386. https://doi.org/10.1145/3132847.3132925
20. Zhang Y, Lyu T, Zhang Y (2018) COSINE: community-preserving social network embedding from information diffusion cascades. In: 32nd AAAI conference on artificial intelligence AAAI 2018, pp 2620–2627
21. Shchur O, Günnemann S. Overlapping community detection with graph neural networks
22. Ine L (2019) Supervised community detection, pp 1–24
23. Hinton GE, Zemel RS (1994) Autoencoders, minimum description length and Helmholtz free energy. Adv Neural Inf Process Syst 6:3–10
24. Janowski T, Mohanty H (2010) Distributed computing and internet technology: preface, vol 5966. LNCS
25. Perozzi B, Al-Rfou R, Skiena S (2014) DeepWalk: online learning of social representations. In: Proceedings of ACM SIGKDD international conference on knowledge discovery data mining, pp 701–710. https://doi.org/10.1145/2623330.2623732
26. Tang J, Qu M, Wang M, Zhang M, Yan J, Mei Q (2015) LINE: large-scale information network embedding. In: WWW 2015—proceedings of 24th international conference on world wide web, pp 1067–1077. https://doi.org/10.1145/2736277.2741093
27. Huang Y et al (2016) node2vec real-time video recommendation exploration categories and subject descriptors. World Neurosurg 95(1):41–50
28. Tran PV (2019) Learning to make predictions on graphs with autoencoders. In: Proceedings of 2018 IEEE 5th international conference on data science advanced analysis DSAA 2018, pp 237–245. https://doi.org/10.1109/DSAA.2018.00034
29. Li S, Jiang L, Wu X, Han W, Zhao D, Wang Z (2021) A weighted network community detection algorithm based on deep learning. Appl Math Comput 401:126012. https://doi.org/10.1016/j.amc.2021.126012
30. Bhatia V, Rani R (2018) DFuzzy: a deep learning-based fuzzy clustering model for large graphs. Knowl Inf Syst 57(1):159–181. https://doi.org/10.1007/s10115-018-1156-3
31. Bhatia V, Rani R (2019) A distributed overlapping community detection model for large graphs using autoencoder. Futur Gener Comput Syst 94:16–26. https://doi.org/10.1016/j.future.2018.10.045

Heart Disease Prediction Using Modified Machine Learning Algorithm

Bavneet Kaur and Gaganpreet Kaur

Abstract Heart patient number is escalating day by day, and numerous individuals lost their precious lives each year due to sudden heart attack in all over the world. Because of this, before time diagnosis of cardiovascular disease is necessary to prevent death. Some technology-based software is required to help in medical field to recognize heart patients with more accuracy and lesser time. Huge amount of heart patients are present in different hospitals in all over the world, which can be used efficiently to diagnose the heart disease by applying data mining techniques. In the process of data mining, knowledge or useful information is extracted among the large sets of raw data. In the prediction analysis, machine learning techniques are applied to discover valuable patterns and forecast future events or trends. This research work will predict the likelihood of coronary heart disorder in patients by implementing a modified machine learning algorithm. The input data are passed through various procedures comprising preprocessing, clustering, and selection of effective attributes before classification. To determine the heart illness, four algorithms which include random forest, K-means, genetic algorithm, and logistic regression are assimilated. In this technique, the irrelevant attributes of heart dataset are discarded to improve the performance and to decrease the training period time. This process is completed by random forest technique. K-means clusters are optimized by genetic algorithm in order to group all the outlier data points. At last, logistic regression is applied to classify the patients based on the heart disease. Performance comparison among various existing techniques has analyzed on the basis of some performance measures. The calculated accuracy increased up to 95%.

Keywords Data mining · Heart disease · K-means · Genetic algorithm · Random forest · Logistic regression

---

B. Kaur (✉)
Sri Guru Granth Sahib World University, Fatehgarh Sahib, India
e-mail: bavneetkaur2000@gmail.com

G. Kaur
Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab, India
e-mail: kaur.gaganpreet@chitkara.edu.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_16

1 Introduction

Prediction of heart disease is complex and time consuming task. Doctors and medical practitioners use their own experience and knowledge to identify heart patients by performing some medical tests. However, huge amount of heart patients are present in different hospitals in over the world, which can be used efficiently to diagnose the heart disease by applying data mining techniques. In this way, this approach may assist medical professionals to diagnose heart disease and to take the right decision for treatment by increasing their knowledge about heart disease. This research work will analyze different data mining classification techniques that are used for heart disease prediction, clustering techniques, and optimization techniques. Also, a new modified heart disease prediction algorithm will be created by combining the better techniques to get the better and accurate results. Firstly, the input data are processed manually, and then clustering of input data is performed by using K-means clustering algorithm. Secondly, these clusters will be optimized by using genetic algorithm to group any outlier data points. Thirdly, clustered data are passed through random forest to separate relevant features from irrelevant features. So, final output performance can be improved. Finally, classification of heart disease patients and non-heart disease patients is done through logistic regression. The output performance will be analyzed in terms of various parameters.

Data mining is the phase which is involved in the phenomenon of knowledge discovery data (KDD). Moreover, data mining is the process in which hidden patterns are identified and converted into more significant data [1]. Data mining is the operation of obtaining the useful decision forming patterns among the historical records which are used to predict the future outcomes [2]. In data mining, the integration of statistical analysis, database technology, and artificial intelligence is used to identify or uncover the covered patterns and relationships from bulky databases. Two techniques such as supervised and unsupervised learning are used to mine knowledge from raw data [3]. Several industries believe data mining is useful to generate profits in business. The purpose of each data mining technique is different based on the modeling objectives like prediction and classification. Classification models generate categorical input, and prediction models generate continuous data as output [4]. Predictive analytics is the technique in which prognostication of upcoming occurrence is done. In addition, the use of data mining in medical field has become popular in the recent years because it predicts diseases on the basis of clinical data of patients. Artificial intelligence techniques can support medical practitioners and professionals for early diagnosis of diseases [5, 6]. In the recent years, health of humans is degrading day by day due to developments and advancements in various fields. People are suffering from chronic cardiovascular disease, and this leads to high-mortality rate. Data mining in healthcare field can be used to forecast the patients future history based on their history [7]. In this way, the risk of an increasing number of health complications due to a chronic disease can be predicted. The complications associated with arteries, heart and circulatory system, inflammatory heart, and congenital heart are known as the cardiovascular heart disorder [8, 9]. The major reason of deaths in most of the developed as well as developing countries is cardiovascular disease as stated by World Health Organization (WHO). It is also the main reason of deaths in adults in all over the world. In heart disease, infection may pass to different areas of heart itself due to which different terms are used to refer them. The major forms are congenital heart problem, heart arrhythmia, cardiopulmonary arrest, cardiovascular disease, coronary heart disease, heart failure, heart muscle complication, pulmonic valve stenosis, etc. When heart beats with irregular or unusual pattern are known as arrhythmia. Ischemia is defined as the blockage of blood vessels due to some hard material like fat [10]. Coronary heart disease (CHD) incorporates myocardial infarction, angina pectoris, and chest pain. Myocardial infarction is also known as heart stroke or heart attack. Chest pain, breathing difficulty, dizziness, slow heartbeat, fast heartbeat, etc., are some common symptoms of this disorder. The usual indication of heart disorder is tenderness in chest. These disorders occur when any coronary artery which supplies blood and oxygen to the heart is blocked and results in inappropriate blood supply to heart. Sometimes, narrowing of coronary artery leads to inadequate oxygen and blood supply to heart, which leads to heart disease or heart attack. The primary reason of heart disease is modern sedentary lifestyle which instantly climbs the levels of blood pressure, diabetes, cholesterol, and obesity. Moreover, some genetic heart diseases are there such as hypertrophic cardiomyopathy, which is gene deformity [11]. In medical field, physician examines the patient based on the personal details and patient's medical history. It requires lot of hard work and time consuming process. On the other hand, data mining techniques can assist the medical profession to take a decision about patient's treatment [12]. Nowadays, the role of data mining techniques in healthcare field is quite popular because it predicts the likelihood of any disease by identifying the new patterns from medical history of patients. An increasing number of scientist avail data mining methods to find the probability of heart disease. Number of scientists and researchers examining the application of all these techniques in order to diagnose the diseases precisely. However, the results of these techniques vary [13]. The data mining-based heart disease prediction automation system decreases the cost as well as time of analyst and patients [14].

2 Literature Survey

Anjan Nikhil Repaka et al. (2019) proposed a technique to determine the heart disorder by examining the old data and information [15]. Threats associated with heart disease are determined by a system known as smart heart disease prediction, which uses Naïve Bayes model. Due to improvement in technology, mobile healthcare technology has plunged to a great extent. The most relevant features are collected from the previous medical data after organizing the essential data into a standard format. The data are secured by applying advanced encryption standard (AES). The data excavation approach used for coronary illness prediction also discussing numerous knowledge abstraction methods. The output generated by smart heart sickness prediction model shows that it predicts the chances of heart disease effectively.

Ankita Dewan, et al. (2015) presented effective hybrid scheme in which genetic algorithm and back propagation technique are combined for heart disease prediction [16]. The major idea is to construct a framework to find unidentified patterns and relations from the historical records of heart patients for early detection of this ailment. This proposed system can cater professionals in medical field to make smart decisions by resolving complicated queries associated with heart disease. Finally, this technique generates efficient results which leads to less medical cost for treatment.

Monika Gandhi, et al. (2015) elaborated various knowledge abstraction techniques with the help of knowledge discovery approaches which are generally used for heart illness prediction [17]. The author realized that enormous data are produced everyday in healthcare field but there is no appropriate method to obtain advantageous information from these records. Data mining is the best solution for this problem. Therefore, various data mining methods can be applied to accomplish the target. All the available data mining methodologies are analyzed and implemented on ancient medical data to obtain useful information. These techniques are presently used for prediction of various diseases.

Rashmi G Saboji, et al. (2017) in this work, some relevant features are taken from large set of clinical databases to predict the heart disease [18]. The proposed work implemented random forest technique on Apache Spark. Medical professional utilized this solution on wide range of medical databases which diagnosed the heart disease with accuracy of 98%. The similarities and dissimilarities between Naïve Bayes classifier and random forest have discussed. The outcomes reveal that random forest outperforms the Naïve Bayes algorithm.

T. Peter, et al. (2012) proposed the usage of data mining methods and pattern recognition in danger detection models in the cardiovascular domain [19]. Due to the natural linear relationship of variables in the input data, the traditional medical scoring systems are not proficient to model non-linear patterns within medical data. This limitation of traditions systems was solved by applying classification techniques which can completely detect complicated non-linear patterns among the dependent and independent variables. Classification data mining models generally classify the data into different categories. Also, these models were capable to detect the relationships between predictor variables.

Cincy Raju, et al. (2018) analyzed that heart disease is the fatal disease which leads to death. Heart disease also causes a critical disorder which remains for many years [20]. It attacks man suddenly. Therefore, early diagnosis of heart disease is highly demanding to support health professionals. The major problem in biomedical is the accurate diagnosis of heart disease. The principal goal of this article is to develop effectual solution to this problem by utilizing using data mining technology. The implementation of various data mining methods like decision tree, neural network, Bayesian classifier, support vector machines, association rule, and K-nearest neighbor classification was done for the heart disease prediction. The results reveal that SVM outperforms the rest of classifiers.

Aakash Chauhan, et al. (2018) identified the rule to forecast the risks of having heart infection or complication in patients [21]. The author analyzed that the data mining technique weighted association rule mining can be used to eradicate manual task to find knowledge from large databases of medical records. This technique extracts the useful information directly from the electronic medical records and helps to reduce the expenses of services and save lives. There was high requirement of these techniques because many people relied on the healthcare organizations for the prediction of cardiac disease. People need results with high accuracy in reduced time. The results of this experiment revealed that large set of rules helps to predict the coronary heart disease with greater accuracy.

M. A. Jabbar, et al. (2016) estimated that the primary reason of deaths in all over the word is coronary heart disease. There is requirement of automated support system to take decisions in detecting the heart disease [22]. The assumption of conditional dependency in conventional Naïve Bayes is reduced by hidden Naïve Bayes (HNB) classifier. In proposed model, HNB was experimented to classify heart patients. The test results show that the hidden Naïve Bayes (HNB) achieved accuracy up to 100% which was better than traditional Naïve Bayes when applied on heart disease dataset.

Imran Mirza, et al. (2019) emphasized for implementation of data analysis to assist the medical professionals in all over the world. The main purpose is to support medical practitioners with useful knowledge and patterns so that they use them if required [23]. The author expecting to use machine learning methodologies to mine patterns in medical databases in order to achieve accurate prediction of cardiac disease in patients. The attributes which highly affect the diagnosis of heart complication are identified in this paper. To classify individual's on the basis of heart disease and to analyze the performance, radial basis function kernel SVM, linear SVM, Naïve Bayesian classifier, and K-nearest neighbor are implemented. It used Boolean values 0 (no heart disease) and 1(heart disease).

3 Research Methodology

Numerous individuals lost their precious lives each year due to sudden heart attack in all over the world. This disease has started attacking the age group 25–40 in the past ten years which leads to sudden death of an individual. The major cause of heart disorder is modern sedentary lifestyle, and sometimes, genetic disorders are also responsible. Therefore, early diagnosis of heart disease is essential so that life loss can be reduced. To achieve this, knowledge mining techniques have been utilized by many researches to excavate useful information and patterns from huge medical dataset. Many data mining schemes are being enhanced or combined to boost the results of heart disease prediction with regard to correctness (accuracy). In proposed work, hybrid combination of machine learning approaches is applied on heart patient's dataset. This process involves few steps which are given below:

3.1 Data Acquisition

In data acquisition, the past data of heart patients are fetched from several medical databases to implement data mining algorithms. In this work, the input is gathered from Cleveland heart disease dataset from UCI machine learning repository to evaluate modified machine learning algorithm. Cleveland processed dataset contains 303 instances and 13 predictor variables. It has 5 classes (0 indicates no heart disease and 1, 2, 3, 4 indicates heart disease)

3.2 Data Preprocessing

To obtain better results of machine learning algorithms, input data should be clean and complete. This step is applied to remove any noisy data and handle missing and redundant values so that the algorithm generates effective outcomes. This research used mode of the present values to fill missing values.

3.3 Feature Selection

This step extracts the most relevant features in diagnosing heart disease rather than using the all attributes present in the dataset. This is done to improve performance of machine learning methods. Various techniques help to select few features from the dataset. In the proposed method, in order to discard less relevant features and select most relevant attributes, random forest technique is implemented so that training time can be reduced to some extent. The random forest algorithm generates a forest of tree structure of the most effective attributes by taking guess score of 100. Random forest classifier chooses those features which are most appropriate for heart disease prediction. Before this step, segmentation of input data is performed by availing K-means clustering technique. It helps to remove outliers data points in dataset. The K-mean clustering is applied which can cluster similar type of information. It is a simple and extensively utilized clustering algorithm. For an image of N pixels, the K-means algorithm focuses on dividing the image into K clusters at which the user offers the value to K. Instead of employing the position of pixels in picture, pixels are grouped on the bases of their values, till the position is presented as a defined feature. Assume a collection of x_i points as $X = \{x1, \ldots, xN\}$ and the feature vector connected to point is represented with $V(x_i)$.There are several stages of K-means technique.

Initialization of Parameters: The initialization of parameters of each cluster is done by possible feature values. The traditional K-means algorithm selects the initial centroid values randomly from all available values. To demonstrate, in case, the property vector presents in the form hue, saturation, and intensity value (HSV), the selection of first element H is done at random from all possible hues.

Allocation of Points to Clusters: When every K clusters C_k consists of a mean μ_k, a distance K function which assists in evaluating the distance amid two property vectors is utilized to assign each pixel x_i to the cluster having the closest mean. Here, each pixel x_i is related to single cluster C_k.

Parameter Recomputation: This property is focused on recomputing the clusters on the basis of attribute values of all data points in every cluster.

Therefore, μ_k is evaluated as the mean of $\{V(x_i)|x_i \in C_k\}$.

Iterate through step 3.3.2 and step 3.3.3 until the movement of points among clusters has stopped. The genetic algorithm is applied on the K-mean algorithm which can optimize the generated clusters of K-mean algorithm. Genetic algorithms (GA) are biologically inspired algorithms. These algorithms represent a novel computational model with its roots in developmental sciences. In general, genetic algorithms refer to optimization process in a binary search space. In contrast to the conventional hill climbers, these algorithms do not assess and enhance a solo solution but a set of solutions or hypotheses known as population. These algorithms generate successor solutions by mutating and recombining the optimal existing identified hypotheses. Therefore, offspring of the fittest solutions replaces a part of the current population after every iteration process. In particular, the searching for a space of candidate hypotheses is carried out for identifying the optimal solution. This solution is termed as the optimization of a specified mathematical measure.

3.4 Classification

In this final step instead of class labels, the formed clusters and selected features are given as input to logistic regression model. There are two clusters: first representing the patients having heart disease and the other is for the individual's with no heart disease. The logistic regression model is trained. During testing, the model will classify the patients into two categories that is yes which indicates that the patient has heart disease, and no class indicates that the individual has healthy heart based on the chances of heart disease.

3.5 Proposed Algorithm

```
Input: Dataset of Heart
Disease
Output: Classified Data
1. K-Mean Clustering (Input Dataset)
Initialize the value of K (Number of clusters)
1.2. Define the centroid of K clusters randomly Repeat while (Data Get Clustered)
Allocate each point to its nearest cluster on the basis of centroid
Genetic Algorithm (Centroid of clusters)
Repeat
while (optimal centroid)
Calculate fitnessvalue 1.3.1.2
Calculate Crossover
If (there are some chromosomes)
1.3.1.2.1 Generate
Mutationsend if
1.3.1.3 Generate Optimal centroid point end while
1.4. Generate Clusters based on distance between points and centroid end while
2. Random Forest for Feature Selection (Clustered Data)
Randomly separate 'K' features from total 'M' features such that K less than M
Split the best attribute 'A' based on its decision capability for the node 'N'
Divide the node 'N' into successor nodes, for each value of A
Iterate the steps 2.1 to 2.3 until accurate classification has been achieved
Generate and grow numerous trees by repeating step 2.1 to 2.4
for 'n' times, where 'n' is the guess score for generating trees.
3. Logistic Regression (Extracted Features)
Repeat for 1 to Number of extracted features
Generate a vector Y which takes binary value 3.1.2Assign
Y=1 if it relates to class label otherwise Y=0
3.1.3. Apply Logistic Regression to X (Input) to find Output
```

4 Results and Discussion

In the proposed work, Cleveland processed cardiac disease dataset is accumulated from the UCI repository to perform experiments. Cleveland dataset has 14 features which include age, gender, cholesterol, blood sugar, number of vessels colored, thalassemia, etc. The performance of proposed algorithm is examined with regards to accuracy, precision, recall, and execution time by implementing it on dataset for forecasting the chances heart disease. Then, the performance of proposed machine learning algorithm is compared with decision tree, multilayer perceptron, Naïve

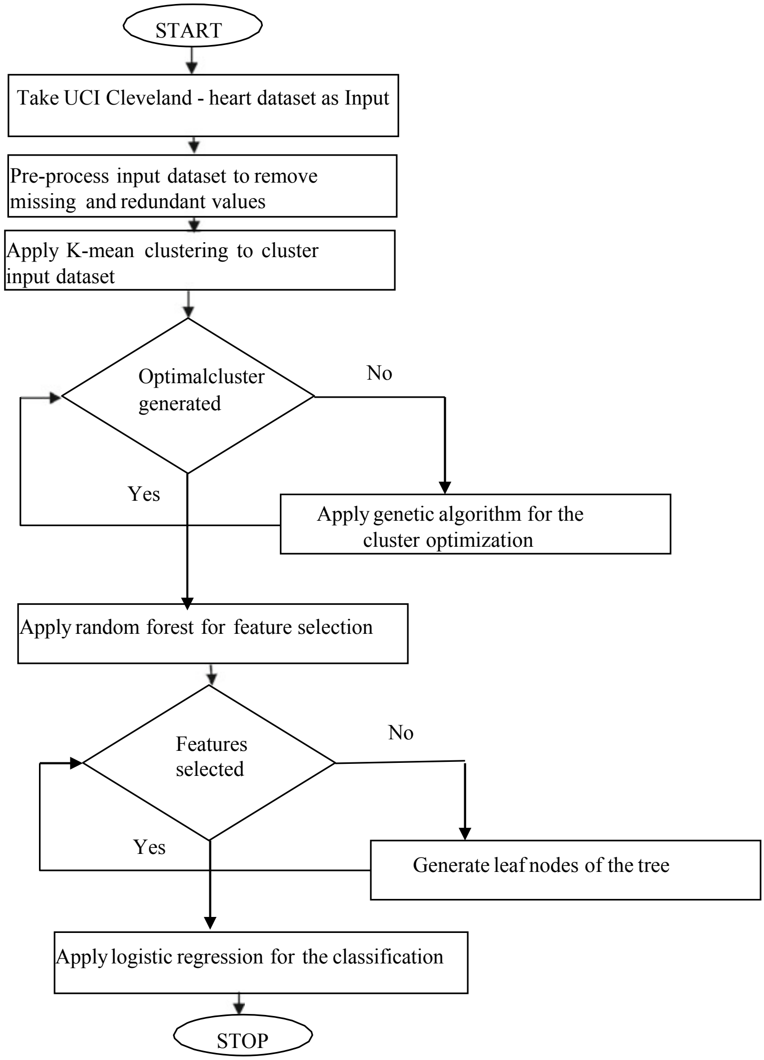


Fig. 1 Proposed methodology

Bayes, and ensemble voting classifier in which Naïve Bayes, multilayer perceptron, and random forest are combined.

Accuracy

Accuracy is the ratio of total number of correctly classified instances to the total number of available instances for a program *i*.

Table 1 Performance analysis

Performance measure	Results
Accuracy	95.08
Precision	95
Recall	95
Execution time	0.8 (sec)

Table 2 Performance analysis by using dataset variations

Performance metrices	Preprocessed dataset	Clustered dataset (K-means)	Feature selected (random forest)	Proposed model
Accuracy	85.71	94.50	87.91	95.08
Precision	86	94	88	95
Recall	85	94	88	95

Precision

Precision is defined as the ratio of number of correctly classified positive samples and the total instances classified as positive. Precision is also known as the positive predicted value. Precision is presented in terms of correctly classified values to the total values that are classified and carry a positive value.

Recall

The recall is the division of correctly classified positive samples with the total positive instances. The recall is also known as sensitivity. The recall is relied on a particular class, i.e., class yes (1) or class no (0).

Execution Time

It is defined as the time consumed by an algorithm to execute. In other words, the time during which a program is running.

The performance analysis of proposed model is represented in Tables 1 and 2. In Table 2, some variations in dataset are used such as preprocessed dataset, clustered dataset, and feature selected dataset to evaluate the performance of logistic regression.

Table 2 illustrates the performance of logistic regression when combined with other data mining techniques. It is clear that logistic regression has achieved high accuracy when applied on clustered dataset rather than on dataset which has most relevant features. On the other hand, proposed model has high accuracy than other variations.

4.1 Performance Comparison

To analyze the proposed model in detail, it is compared with the implementation of other classifiers includes decision tree (C4.5), voting classifier, Naïve Bayesian, and multilayer perceptron, on the same dataset.

Figure 2 denotes the comparison of performance of decision tree (C4.5), voting classifier, Naïve Bayes, multilayer perceptron, and proposed algorithm. It is analyzed that the proposed method performed better with regard to precision, recall, accuracy, and execution time for predicting the heart disease.

Table 3 Performance Comparison

Models	Accuracy (%)	Precision (%)	Recall (%)	Execution time (Sec)
C4.5	75.20	75	75	1.4
Simple Naïve Bayes	83.61	84	84	1.21
Voting classifier	85.25	86	85	1.1
Multilayer perceptron	83.61	85	84	1.3
Proposed algorithm	95.08	95	95	0.9

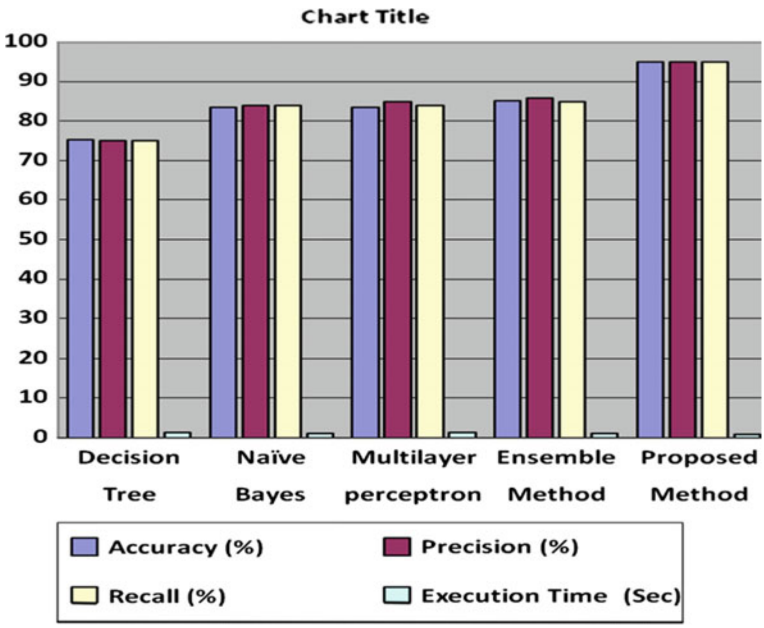


Fig. 2 Comparative performance analysis of all models

5 Conclusion

Heart attack is the major reason of deaths in adults over the recent years. There is high demand of knowledge mining and machine learning-based heart disease detection system so that it can assist practitioners and professionals in medical domain. It is analyzed that heart disease forecast includes several of attribute due to which it is very difficult for machine learning algorithms to achieve higher accuracy. Therefore, it is necessary to remove irrelevant features from the dataset. In proposed model, random forest, logistic regression, and improved K-means algorithms are integrated to predict heart disorders. The K-mean clustering used for the clustering of the data so that outliers can be removed and improve the performance of logistic regression. The K-mean clusters are optimized using genetic algorithm. Random forest is used to extract most relevant features, and classification is done by applying logistic regression. More powerful algorithm can be developed for high accuracy by using other better clustering methods and nature-inspired optimization algorithms. Recent data of patients that experienced cardiac arrest can be collected from hospitals and other sources to make a powerful heart disease prediction system.

6 Future Scope

There is more improvement in devised approach is possible by applying various deep learning approaches. It is possible to make improvement in the new algorithmic approach by comparing it with other existing heart disease prediction algorithmic approaches. Another possibility is that this algorithm can be developed as Web application so that everyone can take advantage of this artificial intelligence-based heart disease forecast system.

References

1. Gomathi K, Shanmugapriyaa D (2016) Heart disease prediction using data mining classification. Int J Res Appl Sci Eng Technol 4(2)
2. Benjamin Fredrick David H, Antony Belcy S (2018) Heart disease prediction using data mining techniques. ICTACT J Soft Comput 9(2)
3. Vijayashree J, Sriman Narayana Iyengar NCh (2016) Heart disease prediction system using data mining and hybrid intelligent techniques: a review. Int J BioSci Bio-Technol 8(4):139–148
4. Ratnam D, HimaBindu P, MallikSai V, Rama Devi SP, Raghavendra Rao P (2014) Computer-based clinical decision support system for prediction of heart diseases using Naïve Bayes algorithm. Int J Comput Sci Inf Technol 5(2):2384–2388
5. Santhanam T, Ephzibah EP (2015) Heart disease prediction using hybrid genetic fuzzy model. Indian J Sci Technol 8(9):797–803
6. Purusothaman G, Krishnakumari P (2015) A survey of data mining techniques on risk prediction: heart disease. Indian J Sci Technol 8(12)

7. Srinivas K, RaghavendraRao G, Govardhan A (2010) Analysis of coronary heart disease and prediction of heart attack in coal mining regions using data mining techniques. In: Proceedings of 5th International conference on computer science & education, China, pp 24–27
8. Peter J, Somasundaram K (2012) An empirical study on prediction of heart disease using classification data mining techniques. In: Proceedings of IEEE international conference on advances in engineering, science and management (ICAESM), pp 514–518
9. Masethe HD, Masethe MA (2014) Prediction of heart disease using classification algorithms. In: Proceedings of the World Congress on engineering and computer science (WCECS), San Francisco, USA
10. Soni J, Ansari U, Sharma D, Soni S (2011) Predictive data mining for medical diagnosis: an overview of heart disease prediction. Int J Comput Appl 17(8):0975–8887
11. Rajkumar A, Sophia Reena G (2010) Diagnosis of heart disease using datamining algorithm. Global J Comput Sci Technol 10(10)
12. Soni S, Vyas OP (2010) Using associative classifiers for predictive analysis in health care data mining. Int J Comput Appl 4(5):33–34
13. Ansarullah SI, Sharma PK, Wahid A, Kirmani MM (2016) Heart disease prediction system using data mining techniques: a study. Int Res J Eng Technol (IRJET) 3(8)
14. Bhatla N, Jyoti K (2012) An analysis of heart disease prediction using different data mining techniques. Int J Eng Technol 1(8)
15. Repaka AN, Ravikanti SD, Franklin RG (2019) Design and implementing heart disease prediction using Naives Bayesian. In: 3rd International conference on trends in electronics and informatics (ICOEI)
16. Dewan A, Sharma M (2015) Prediction of heart disease using a hybrid technique in data mining classification. In: 2nd International conference on computing for sustainable global development (INDIACom)
17. Gandhi M, Singh SN (2015) Predictions in heart disease using techniques of data mining. In: International conference on futuristic trends on computational analysis and knowledge management (ABLAZE)
18. Saboji RG (2017) A scalable solution for heart disease prediction using classification mining technique. In: International conference on energy, communication, data analytics and soft computing (ICECDS)
19. John Peter T, Somasundaram K (2012) An empirical study on prediction of heart disease using classification data mining techniques. In: IEEE-International conference on advances in engineering, science and management (ICAESM)
20. Raju C, Philipsy E, Chacko S, Padma Suresh L, Deepa Rajan S (2018) A survey on predicting heart disease using data mining techniques. In: Conference on emerging devices and smart systems (ICEDSS0)
21. Chauhan A, Jain A, Sharma P, Deep V (2018) Heart disease prediction using evolutionary rule learning. In: 4th International conference on computational intelligence & communication technology (CICT)
22. Jabbar MA, Samreen S (2016) Heart disease prediction system based on hidden Naïve Bayes classifier. In: International conference on circuits, controls, communications and computing (I4C)
23. Mirza I, Mahapatra A, Rego D, Mascarenhas K (2019) Human heart disease prediction using data mining techniques. In: International conference on advances in computing, communication and control (ICAC3)

Entrust SDP Authentication to Software-Defined Campus Network (SDCN)

Suruchi Karnani and Harish Kumar Shakya

Abstract At the beginning of the fall of 2020, the campus network poses new challenges due to the outbreak of the COVID-19. Entire entities of campus are scrambled to set up remote learning. In turn, the number of users and devices on the network multiplies in a tremendous way. This enlargement forces network administration to control and verify the accessibility. To address this subject, this paper puts forward the software-defined perimeter (SDP) integrated with the software-defined campus network (SDCN) framework. SDP controller is united with SDN controller in SDN control plane to yield authentication and access control for network. SDCN with SDP provides strong prospects in minimizing unauthenticate access ratio which strengthens the trust factor of legitimate users and enhances quality of service. SDP with SDCN helps in enabling additional boundaries within the network which acts as a defense, enhances scalability, and shields the network from external attacks as well as from internal malicious users.

Keywords Authentication · SDN · SDP · Campus network · AAA

1 Introduction

We all are experiencing the rise of network devices on the Internet. More or less 100% of enterprises, campuses, and universities are dominated by network devices [1] in the COVID-19 pandemic. On campuses, devices are thronging onto Wi-Fi rapidly and frantically. Hence, the network intricacy upheaved in no time. To manage current demands, the environment needs an architecture with automation, scalability, and agility features. SDN paradigm caters to the above-mentioned features and provides programmability which is highly required to govern the current network activities

S. Karnani (✉) · H. K. Shakya
Amity University, Gwalior, Madhya Pradesh, India
e-mail: suruchi.karnani@s.amity.edu

H. K. Shakya
e-mail: hkshakya@gwa.amity.edu

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_17

and user demands. The central logic of the SDN controller acts as a network operating system, and the OpenFlow communications interface between the controller and the underlying infrastructure is open standard software. TLS encryption is an optional feature in SDN [2]. Security measures are not the in-built feature of SDN, and this may severely stir the authentication, security, and integrity of the network. Hence, security and authentication (trust) are essential strands to consider from the academic and industrial points of view [3]. However, the existing security approaches are measured, and it is prominently observed that a new paradigm is required to shield enterprises proactively.

An analogous framework is the software-defined perimeter (SDP) which offers modular, agile, and secure infrastructure similar to SDN [4]. SDP focuses on securing the user, the application, and the connectivity in-between as well as different levels of access to different users which cannot be provided by virtual private network (VPN). Different level of access by the different user is highly required by SDCN and business enterprises. To seek zero-trust security, the adoption of SDP technology has grown in enterprises. It makes application infrastructure invisible to the Internet, so it evades network-based attacks (flooding attacks, DoS, ransomware, malware, server scanning) [5], and it narrows down the enterprise and campus threats. Thus, to secure the SDCN environment, an SDP-enabled authentication approach is proposed in this paper. Therefore, contributions of this paper can be outlined as follows:

- Propounded the idea of amalgamating SDP with SDCN to ensure authentication and integrity. In this approach, legitimate and role-based user can access the infrastructure as specified access control list.
- SDN controller gets free from re-authentication login system. Re-authentication logins are handled by SDP gateways.

The remainder of this paper is organized as follows. Section 2 discusses the existing work of SDN integrated with authentication protocol for governing access and authentication. Section 3 explains the preliminary concept of communication between OpenFlow switch and SDN controller and SDP architecture. Section 4 presents our proposed SDCN architecture integrated into SDP, and finally, the conclusion is provided in Sect. 5.

2 Literature Review

In this section, we will shed light on existing works of integrating security frameworks such as 802.1x, Kerberos, and multi-level authentication protocol with SDN to enhance security and SDN-based campus network authentication which is the main focus of this paper. Table 1conveys a brief overview of existing works.

Mutaher et al. [6] put forward Kerberos authentication protocol to ensure the authenticity of the hosts for securing SDN controller. The proposed approach is simulated with AVISPA for verification. They emulate the MITM attack to show the result. The author [7] proposed two-level authentication. Authentication iu imple-

Table 1 Comparative analysis of existing work

References and year	Authentication framework	Authenticate to	method(s)	Remarks
[6], 2021	Kerberos	Host	Verify ID and passwords using centralized server	Stream authentication process connecting network devices, assuring data integrity and privacy
[7], 2020	802.1X with RADIUS and two-level network access	User and a device		
[8], 2019	802.1x (hostpad)	Flow-based and access-based control	Attributes-based access control and particle swarm optimization for secure path	Proactive defense architecture, low response time
[9], 2018	802.1x standard and distributed reactive stateful firewall		Reactive mode, policies are dynamically enforced	Enhances overall performance.
[10], 2017	IEEE 802.1X with RADIUS server and database	Host based	Session database enhances the capabilities of RADIUS	Approach can be scaled, less security concern w.r.t infrastructure
[12], 2016	IEEE 802.1X with RADIUS server	Port-based	Devolve intelligence to switches	Minimizes a load of SDN Controller with low latency
[13], 2015	SDN-enabled VNF built on IEEE 802.1x	Flow-based network access control	VM processes the traffic to decide authentic or unauthentic flow and	NFV deployment supports cost-effective services

mented using 802.1X protocol with a RADIUS server and achieved granularity and real-time monitoring in authentication method. In the paper [8], the author proposed an E-ABAC access control method to secure path planning by the PSO algorithm. The experiment concluded that selecting the parameters is a crucial factor for implementing the access control method.

In the paper [9], the proposed method is a blend of two modules: the network authentication and access control system to guard the network control (802.1x standard) and the distributed firewall system to protect data transmission (policies). This method may work in proactive and reactive mode as well as it builds a boundary line within a network that guards the network against external attacks. In the paper [10], the author proposed a controller application called the authentication and authorization module (AAM). The approach adopts the 802.1X for the authentication process. It maintains the session database of authenticated and authorized users. The author suggested implementing their approach in a distributed way in the network. Diogo Menezes et al. [11] proposed an authentication between hosts and servers named AuthFlow mechanism using layer 2 protocols. IEEE 802.1X standard is applied for authenticity to the hosts with the RADIUS server. Extensible authentication protocol (EAP) is used to encapsulate the messages that are passing among the RADIUS server and the hosts. AuthFlow works under the SDN controller in the application layer. The author [12] put forward the SDN architecture with IEEE 802.1X port-based authentication to enhance security. Approach uplifts the scalability by entrusting the access potentially to the network elements. Hostapd acts as an authenticator for further authentication requests in the POX controller. The proposed approach minimizes the controller's workload and raises the overall performance as well as protects the network with DoS attacks. Matias et al. [13] present flow-based network access control (FlowNAC) to authorize legitimate access into the network. FlowNAC uses the disassociating role of the policy enforcement point. Here, the PEP consists of three elements: SDN datapath, authenticator network function, and SDN controller. The approach helps in extending the legacy switch by EAPOL-in-EAPOL. The main objective of FlowNAC is to perform fine-grained FlowNAC. In comparison with IEEE 802.1X port-based authentication, the proposed approach delays can be longer.

3 Preliminary Concept

In this section, the basic idea of the SDN and the SDP work process is described, to understand how the proposed technique mingled to ensure SDCN authentication.

3.1 OpenFlow Switch and SDN Controller, IH Communication

A diversity of networking trends has contributed to the key role of SDN. Automated workflows, centralized control, network visibility, and abstract control are the prominent features of the SDN. The data layer and the control layer are disjoint. Data layer and control layer communicate through a southbound interface, whereas northbound interface communicates through API applications [14]. Both the infrastructures have common features such as dynamic management, layer abstraction, and centralized

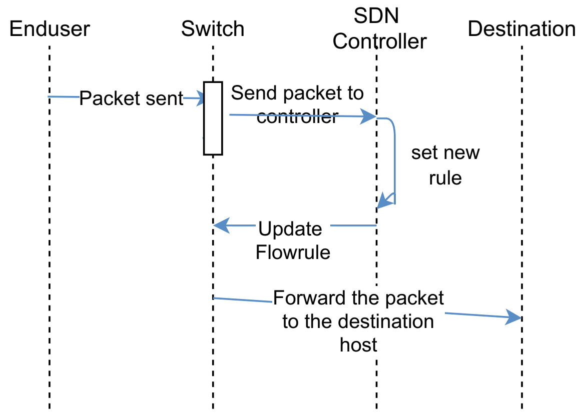


Fig. 1 Switch and SDN controller work process flow

management, and both serve the heterogeneity of devices. Thus, blending campus network with SDN and SDP will turn into complete current network requirements. OpenFlow switch furnishes consistency in data flow management and scalability in the SDN environment. OpenFlow switch consists of a flow table, secure channel, and OpenFlow protocol [15]. An OpenFlow switch communicates with a flow table and a controller via OpenFlow protocol. The process flow between switch and SDN controller is shown in Fig. 1.

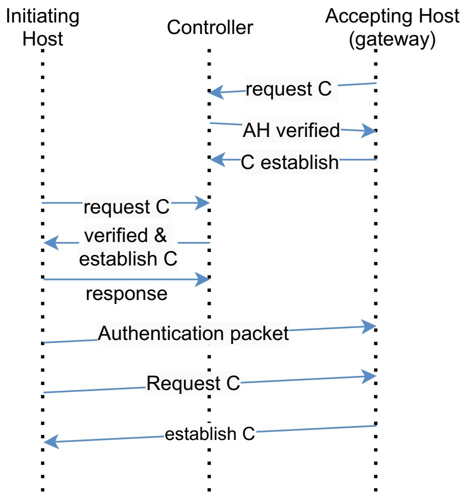
1. Initiating host sent the packet to the switch.
2. Switch will check for the rule match.
3. Switch forward the packet to the controller if a match does not found.
4. Controller sets the new rule and sends it to the switch.
5. Switch will update the rule in the flow table.
6. Then switch will forward the packet directly to the destination host (AH).

3.2 SDP Framework

An SDP architecture consists of an SDP controller, SDP host (initiating host [IH], accepting host [AH]) such as a router, gateway, servers, and switches components. A controller component functions as the centralized policy enforcement engine triggers the authentication action and keeps track of the users, applications, and devices. The end users do not have the ability to see IP or DNS entries of internal resources, and this feature protects the infrastructure from a variety of external attacks [16, 17]. The process workflow of IH, AH, SDP controller, and gateway is shown in Fig. 2

1. Connects SDP to network for authentication.
2. Gateway sends a request to the controller.
3. Gateway verifies and establishes the connection.

Fig. 2 SDP work process flow



4. Gateway updates with all information by the controller.
5. IH initiates a connection to the controller.
6. Connection verifies and established.
7. IH sent verification to the gateway.
8. Gateway establishes the connection between IH and AH.

4 Proposed Architecture

The main idea of this paper is to control access to resources based on identity using a multistage process that limits access to resources only to authorized users. This approach may help to monitor external as well as internal users of the network. The proposed architecture is a conjunction of SDN-based campus network with SDP framework shown in Fig. 3. SDP provides perimeter security to the SDCN. It creates the virtual boundary around the campus network which saves the campus network from turmoils. SDP allows devices to access the specific services and hosts that are permitted by policy, thus it reduces port attacks by malicious users and MITM attacks. In our approach, authentication process will be served by the SDP controller embedded in the SDN control layer. SDP controller designs the policies rules of data flow. Take two scenarios for the authentication process (stateless and stateful switches). The algorithm 1 authenticating controller and SDP gateway (GW) is stateless and stateful switches, and the algorithm 2 is DataPacketForward to verify and forward packets. Both the authentication scenarios are illustrated in an SDP integrated SDCN authentication flowchart in Fig. 4.

If a rule does not match gateway, follow the drop rule [16]. The end user device gets authenticated before getting connected to the network. In SPA, first device gets

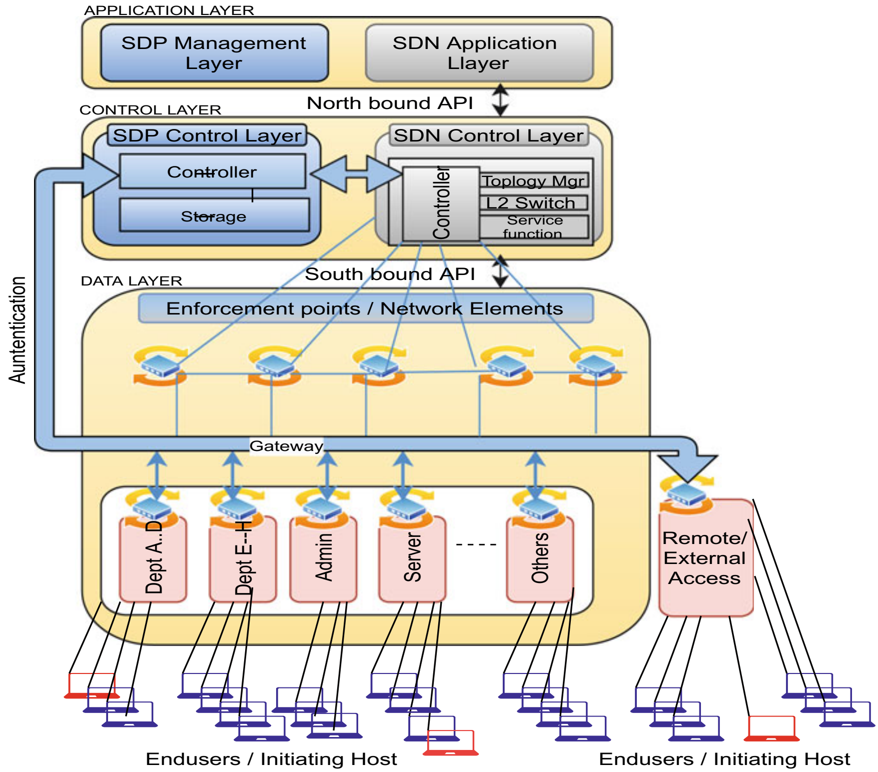


Fig. 3 Proposed SDP with software-defined campus network framework

---

Algorithm 1 Algorithm for Controller and SDP Gateway Authentication

---

procedure C- GW- AUTH(C, GW) ▷ Controller and Gateway
  TLS conn initiated by GW to the C, sends an SPA packet.
  C authenticates GW using certificate
  Establish mTLS secured conn between C and GW
  C send new rule to the GW, GW establish new rule in DB.
  C sends the information to GW for authentication
  if $DB \neq 0$ **then**
    Go to Stateless_Switches.
  else
    Go to Stateful_Switches.
  end if
  `Stateless_Switches:`
  GW updates the DB with new rule and set same FlowRule in switch `Stateful_Switches:`
  GW populates the rules in the Switch flowTable from the DB as per Gw logs. CALL DATA-PACKETFORWARD()
end procedure

Algorithm 2 Algorithm for Data Packet Forward

```
procedure DATAPACKETFORWARD
    EU sents packet
    if FR doesnot exists in Switch Flowtable then
        EU's SPA sent to the SDPC
        SDPC authenticates using certificate
        SDPC establishes connection with EU and send encrypted authentication info to GW.
        EU send another encrypted key to GW
        Packet decrypted and verified.
        if Verified then
            GW Setup the rules and store it in DB
            Update Switch Flowtable with same rule
            Transfer data packets
        else
            Redirect to the Guest Switch and eliminate the FlowRule after a particular time interval.
        end if
    else
        Transfer data packet
    end if
end procedure
```

authenticated, and then SDP authenticates traffic. It authenticates the host at every session. If the flow rule exists in the flow table still, then basic authentication(push button) will verify the connection between source to destination. Initial connection setup time may take a longer time. However, it is resilient to denial of service attacks, MITM, and session hijacking attacks.

5 Conclusion

In SDCN, each and every packet flow communicates with the SDN controller to establish a new flow rule. This paper proposed an SDCN authentication framework with SDP to shield the network from malicious user and threat attacks. In addition, it intensifies the scalability and lessens the probability of single-point failure. SDP verifies host and device identities. This method is applied to strengthen the authentication system of the SDCN infrastructure. And most important point is that the authenticator does not reside inside the network or on the top of the controller. SDP framework is connected to the edge switches only. SDCN with SDP has potential to drop the percentage of un-authenticated access that helps in gaining quality of service and trust of legitimate users. In this paper, we developed the prototype for host and device authentication in the SDCN environment using SDP; in the future, we will develop the implementation bed for the proposed approach concerning the placement of the gateway in infrastructure.

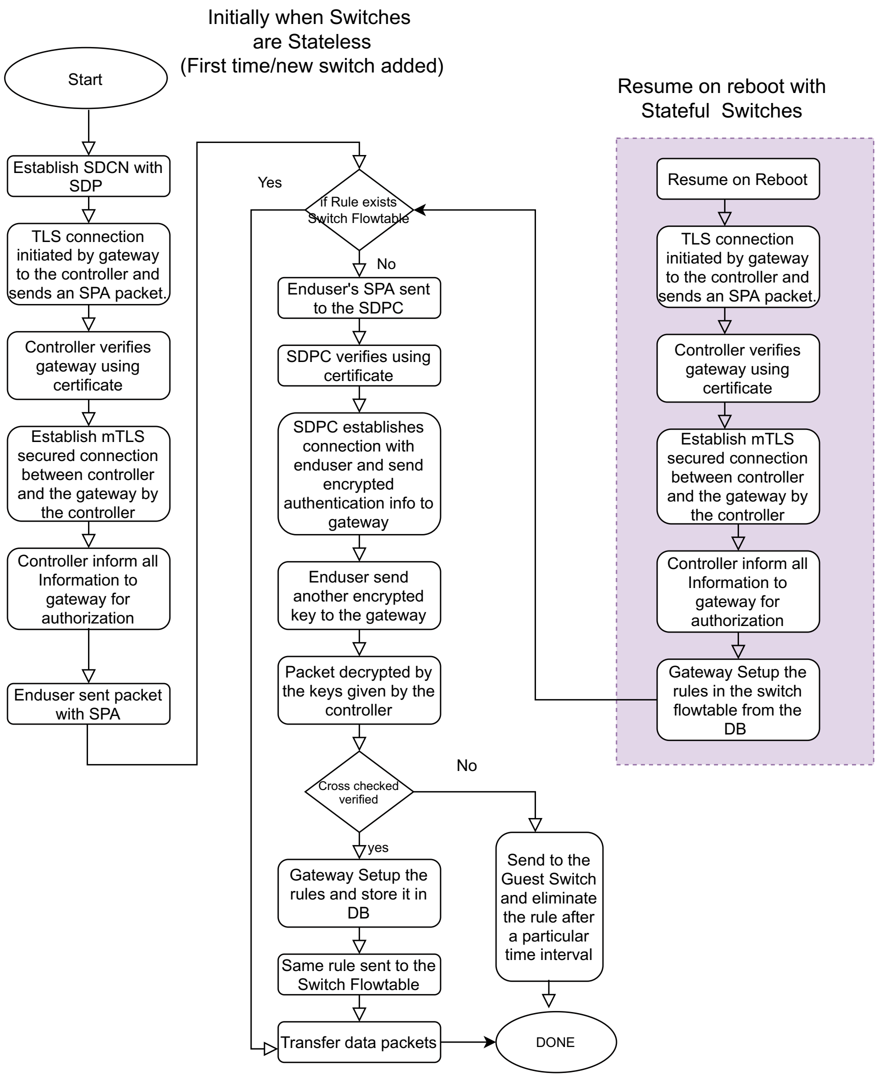


Fig. 4 SDP integrated SDCN authentication flowchart

References

1. https://www.rcrwireless.com/20200218/internet-of-things/connected-devices-will-be-3x-the-global-population-by-2023-cisco-says
2. Benton K, Camp LJ, Small C (2013, Aug) OpenFlow vulnerability assessment. In: Proceedings of the second ACM SIGCOMM workshop on Hot topics in software-defined networking, pp 151–152

3. Scott-Hayward S, O'Callaghan G, Sezer S (2013, Nov) SDN security: a survey. In: 2013 IEEE SDN for future networks and services (SDN4FNS). IEEE, pp 1–7
4. https://www.checkpoint.com/downloads/products/cp-software-defined-protection-enterprise-security-blueprint.pdf
5. Sallam A, Refaey A, Shami A (2019) On the security of SDN: a completed secure and scalable framework using the software-defined perimeter. IEEE Access 7:146577–146587
6. Mutaher H, Kumar P (2021, Jan) Security-enhanced SDN controller-based Kerberos authentication protocol. In: 2021 11th international conference on cloud computing, data science & engineering (confluence). IEEE, pp 672–677
7. Ruixuan P, Chao N, Yingjie Y, Qiang L, Bowen L (2020, Feb) Research on the network access authentication technology of SDN based on 802.1 X. In: 2020 12th international conference on measuring technology and mechatronics automation (ICMTMA). IEEE, pp 780–786
8. Chang D, Sun W, Yang Y, Wang T (2019, Dec) An E-ABAC-based SDN access control method. In: 2019 6th international conference on information science and control engineering (ICISCE). IEEE, pp 668–672
9. Nife F, Kotulski Z, Reyad O (2018) New SDN-oriented distributed network security system. Appl Math Inf Sci 12(4):673–683
10. Hauser F, Schmidt M, Menth M (2017, May) Establishing a session database for SDN using 802.1 X and multiple authentication resources. In: 2017 IEEE international conference on communications (ICC). IEEE, pp 1–7
11. Mattos DMF, Duarte OCMB (2016) AuthFlow: authentication and access control mechanism for software-defined networking. Ann Telecommun 71(11):607–615
12. Benzekki K, El Fergougui A, El Belrhiti El Alaoui A (2016) Devolving IEEE 802.1 X authentication capability to data plane in software-defined networking (SDN) architecture. Secur Commun Netw 9(17):4369–4377
13. Matias J, Garay J, Toledo N, Unzilla J, Jacob E (2015) Toward an SDN-enabled NFV architecture. IEEE Commun Mag 53(4):187–193
14. Kreutz D, Ramos FM, Verissimo PE, Rothenberg CE, Azodolmolky S, Uhlig S (2014) Software-defined networking: a comprehensive survey. Proc IEEE 103(1):14–76
15. McKeown N, Anderson T, Balakrishnan H, Parulkar G, Peterson L, Rexford J, Shenker S, Turner J (2008) OpenFlow: enabling innovation in campus networks. ACM SIGCOMM Comput Commun Rev 38(2):69–74
16. Moubayed A, Refaey A, Shami A (2019) Software-defined perimeter (SDP): state of the art secure solution for modern networks. IEEE Netw 33(5):226–233
17. https://downloads.cloudsecurityalliance.org/initiatives/sdp/Software_Defined_Perimeter.pdf

Preventing COVID-19 Using Edge Intelligence in Internet of Medical Things

R. Mahalakshmi and N. Lalithamani

Abstract Internet of Medical Things (IoMT) is a smart interwoven technology enabled by the advancements made in multi-disciplined fields of medical devices, networking technologies, healthcare applications and artificial intelligence. The current spread of the coronavirus disease (COVID-19) globally has thrown innumerable challenges against human survival. To overcome this pandemic situation, an innovative healthcare solution is vital for saving human lives and mitigating the viral spread. We propose an E-Health+ system that can provide remote patient assistance anytime, anywhere. E-Health+ makes use of artificial intelligence in edge nodes for data processing coupled with Federated learning for swift prognostic medical advice for connected patients during their critical times in IoMT. The medical advice or assistance provided is based on the requests arising in a real-time basis with minimal response times, thereby reducing latency and also the much-needed privacy preservation towards the sensitive patient data.

Keywords COVID-19 · Internet of Medical Things (IoMT) · Remote health care · Edge intelligence · Federated learning

1 Introduction

The novel coronavirus has become the biggest healthcare challenge mankind has faced in the current techno-evolving times. The viral transmission and fatality rate, due to COVID-19, are still rising despite the first case being reported more than two years ago. The viral spread among the world population was attributed to touching an infected surface, coming in contact with the nano-level droplets released in air when an infected person coughs or sneezes resulting in a communal spread. Added

R. Mahalakshmi · N. Lalithamani (✉)
Department of Computer Science and Engineering, Amrita School of Computing, Amrita Vishwa Vidyapeetham, Coimbatore, India
e-mail: n_lalitha@cb.amrita.edu

R. Mahalakshmi
e-mail: r_mahalakshmi@cb.students.amrita.edu

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_18

to that, he or she might be an active carrier throughout the virus incubation period which may last for 5–15 days. So, the spread has been a growing chain around the world as a pandemic, and it is still far from being contained. Though vaccinations are fast paced in countries like India and USA, mutated strains of the virus are still affecting people in different parts of the world. So, precaution and preparedness are the most efficient ways to overcome the pandemic.

Although the prescribed sanitisation and personal distancing measures are helpful in containing the viral spread, the elderly and people with chronic conditions like lung diseases, asthma, diabetes, cancer, hypertension, immune deficiencies, pregnancy, obesity, pulmonary diseases, cardiovascular and neurological problems are comorbid to COVID-19. And such people are at high risk in contracting and spreading this deadly virus. So home quarantining such population can help in containing the viral spread and saving lives. But continuous and timely health assistance should also be provided to these patients based on their day-to-day health conditions.

Internet of Medical Things (IoMT) as a technology is powered by advanced medical devices, networking technologies, healthcare applications, artificial intelligence (AI) and machine learning (ML). The medical devices are built with Wi-Fi capabilities that enable them to make machine-to-machine connections and transfer the sensed data from patients to the medical applications. So, with these technologies at hand, health assistance and medical directives can be provided for the remotely home quarantined, critical-care and aged patients from anywhere, anytime through their hospitals. The vital parametric data from their wearable or mobile phone sensors and health monitors can be analysed continuously to provide unbiased and timely health assistance. This can help them in managing their risk factors along with their illness without hospital visits.

On a technical perspective, the data collected from patients' sensors and health monitors are sent to edge, fog or cloud for analysis which is returned with appropriate remedial action or advice for patients. Cardiac and blood pressure monitors, blood glucose monitors, implanted cardiac defibrillators, pacemakers, respiratory devices, hearing devices, vital sign monitors and oximeters are some of the few connected devices from which the patient data is sensed and processed for analytics.

2 Related Work

AI-empowered machine learning algorithms can detect the interrelationships between the clinical parameters sensed, diagnose the present state of health for the patient and suggest treatments, appropriate health directives. IoT as a technology can deliver e-Health services towards pandemic management [1] using sensors that monitor patients in real time. The study explored the evolution and management phases of IoT and sensor technologies, leading to the state where the current COVID-19 challenges like virus tracing, tracking and migration can be realised using them.

Blueprint for smart connected community scenarios was proposed in [2], which can proactively prevent, control and monitor the COVID-19 epidemic using IoT and data-supported connected environment. The global rise of cases and mortality due to COVID-19 has pushed the research community to seek answers from an interrelated perspective to understand the virus better to root it out.

With the availability of worldwide COVID-19 data from December 2019 to the present, one of the prominent research threads was the impact of comorbidity on people affected by this virus. The study on COVID-19 patients [3] who are comorbid are said to develop severe health complications, which may also lead to death due to the complicated nature of the viral reaction. In a report by Centre for Disease Control [4], three levels of evidences were presented for the different types of underlying chronic conditions and their role in the viral progression. These evidences were the research findings focussed on a single or combination of the chronic diseases on COVID-19.

Higher mortality rate was observed in older patients with Type2 diabetes, and the effect of their medication on COVID-19 was studied [5–7] in China. A study with the help of logistic regression model [8] based on clinical COVID-19 data from Chicago medical centre was used to analyse the risk factors associated with hospitalising the critically ill.

Analysis of risk association for patients with Type I, Type II and other diabetic types [9] was performed with the help of logistic regression for COVID-19 cases in UK hospitals over a 72-day period. The findings concluded that there was an increased risk associated with diabetes, and a third of the total deaths occurred in people with diabetes.

Mortality prediction with the help of machine learning models on patients with COVID-19 in Korea was investigated in [10]. This was based on their combined data from national health insurance Korea and the COVID-19 patient data.

Recent research findings highlight the role of edge computing in reducing the gap or time delay in providing patient care, occurring due to the technical challenges in a connected medical environment. Building an edge computing based on application needs, long-term maintenance routines was outlined in [11], along with the listing of different edge computing platforms.

Computational cost for machine learning at edge with a case study was proposed in [12]. The computational effort was claimed to be reduced by 80% with classification accuracy declining by 3%. Benefits from federated learning for digital health solutions were explored by [13] along with the challenges that still need to be addressed.

Above studies on COVID-19 and other viral diseases like Ebola, SARS and MERS have shown the world that they are capable of creating pandemic situations, as they are silent contagions and are transmitted even before any symptom can appear from an infected person. So, a contactless healthcare service for people has become an irrefutable need in current times.

Smart edge-based healthcare system [14] can efficiently address the healthcare needs on a global level. Computing Resource Allocation strategy for Internet of Medical Things was proposed in [15] that considered energy consumption and the time delay for effective data processing. Federated learning can be the enabling technology in mobile edge computing and this was suggested by [16] as part of their survey on Federated Learning.

Usage of federated learning in a microgrid energy management system was explored in [17] for energy load predictions. Latency minimisation for different offloaded platforms near edge was analysed by [18]. Collaborative data analysis with the help of federated learning in distributed environment was proposed by [19], where the performance achieved showed improvement.

Advanced and rapid growth of smart health technologies involving sensors, robotic process automations, telemedicine products, telehealth services and applications, cloud-based electronic health records, edge computing, medical health data analysis with AI and machine learning and telecommunication connectivity growth has paved the way for achieving contactless, remote health monitoring into a reality.

In this perspective, we propose an E-Health+ system that can provide remote healthcare assistance, which can be complemented along with the government envisioned telemedicine initiatives. Our E-Health+ system proposes a remote healthcare model with a novel architectural setup, data processing capabilities at edge devices and gaining edge intelligence with the help of federated learning. COVID-19 data analysis with machine learning model was performed to demonstrate the power of AI in deriving appropriate inferences from data to provide contactless healthcare.

3 E-Health+

E-Health+ is envisioned as a real-time remote, healthcare system, which can provide health assistance for patients based on their health conditions. Other vital services including the COVID-19 pandemic management, health assistance for aged and critically ill patients. Monitoring, planning and coordination of disease control, immunisation / vaccination for citizens, etc., are envisioned to be part of this system. E-Health+ (Fig. 1) system can be complemented to the existing healthcare network consisting of private and rural healthcare centres, diagnostic laboratories, multi-speciality hospitals and registered pharmacists.

Large amount of data gets generated by sensing the vital health parameters like blood pressure, blood glucose level, heart rate by health monitors, sensors and smart wearable devices worn by the patient. This data is gathered, processed and analysed by our edge intelligence subsystem to provide quick actionable intelligence to patients at the edge nodes, which are closer to them. Thus, gathering and generating intelligence at edge minimises the delays arising in the health network, thereby reducing the latency.

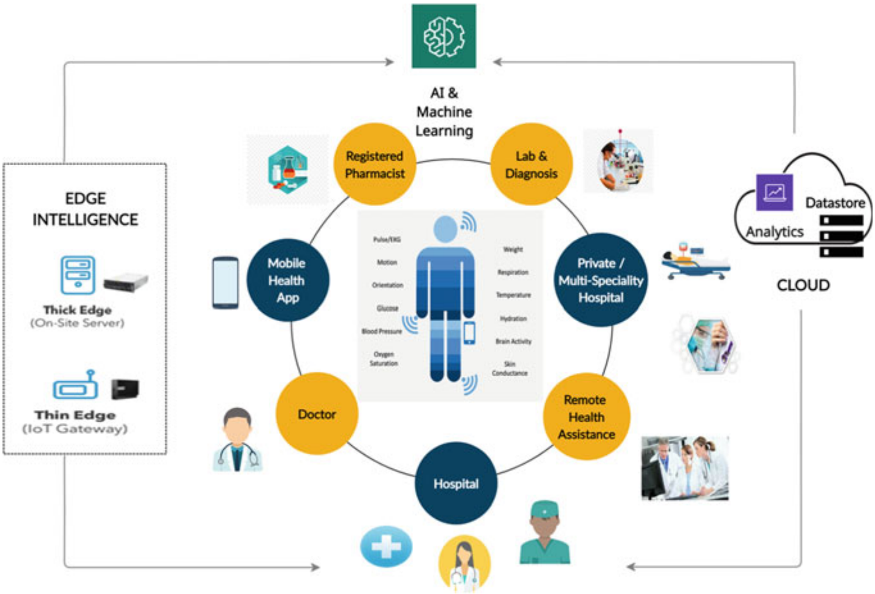


Fig. 1 E-Health+ component subsystems

3.1 Edge Intelligence

Edge intelligence subsystem is envisioned to be part of the existing hospital network consisting of doctors, remote healthcare assisting unit, diagnosis laboratory, pharmacy, hospital in IoMT. It is a decentralised computing architecture where the distributed edge nodes are positioned closer to the patients' sensors/mobile devices and stand connected 24 × 7 to perform data analysis with the help of edge services foreseen. This removes the delay arising due to latency involved in collecting and moving the data from patients' devices to cloud for analysis and bringing the analysis responses back to the patients.

The data processing services in the edge nodes aggregate the patient's data, decide and designate this aggregated data to a suitable node in the network with the help of our resource allocation (RA) framework. The decision for selecting an appropriate node is evaluated on the basis of battery power left and its processing capability, etc. With the appropriate medical intelligence extracted after the analysis, further course of actions in the form of therapy, medicine or other health assistance is suggested through the application or short message by the hospital. So, edge intelligence along with our proposed federated learning approach can provide swift responses to patients in any medical emergencies.

3.2 Federated Learning

Federated learning (FL) is a machine learning technique where a centrally trained machine learning model from cloud is loaded in the edge nodes. Edge nodes here can also be the patient's mobile phone which gets trained and tuned with the new aggregated patient data. Here only the model used for prediction is shared between the edge and cloud, while the sensitive patient data is held in his device, thereby ensuring complete privacy. Figure 2 summarises the FL process which ensures privacy in medical data, lower latency and results in smart predictions, thereby saving time and energy. With appropriate foresight gained in this learning process, the E-Health+ system can provide appropriate medical directive needed by the connected patients.

Machine learning models like neural networks, support vector machines, decision trees, logistic regressors and classifiers can be used in federated learning. The training data for federated algorithms reside at the distributed edges and the model is trained in multiple locations with multiple iterations. This removes the constraint of a large dataset needed to train a model normally. As the model gets trained at edge, data need not be shared, and the trained model gets updated, thereby preserving patient privacy.

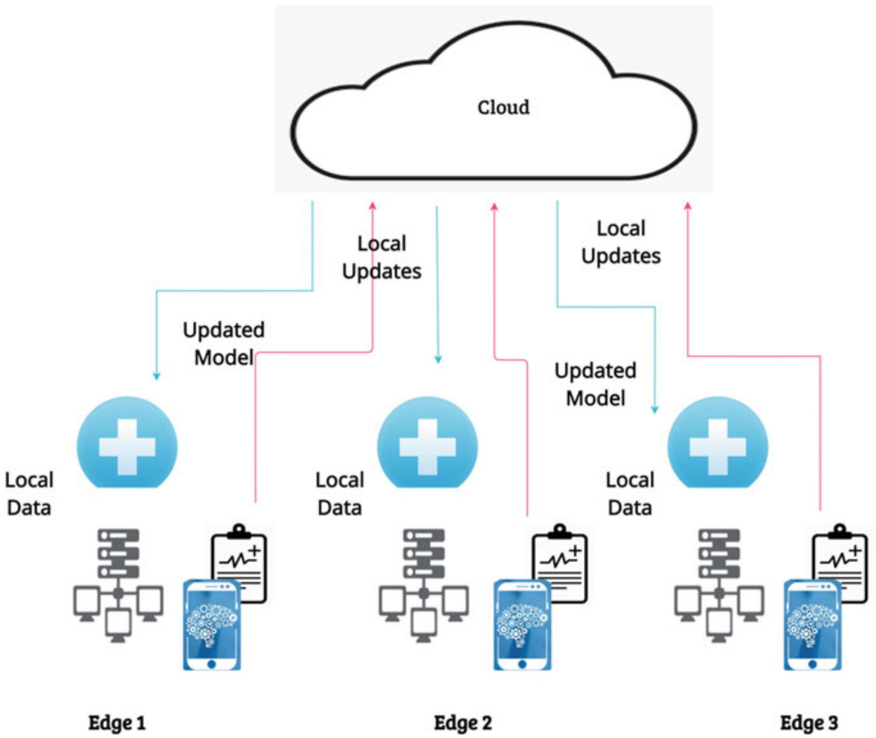


Fig. 2 Federated learning at edge nodes

Data for the device-hosted models are collated from the edge nodes of edge intelligence subsystem. And the data is not independently, identically distributed due to the heterogeneous nature of the devices. A central orchestration mechanism organises the training but will not be accessing individual edge device's data. The steps involved in FL with respect to a model can be summarised as:

1. Initially, the ML model from cloud is distributed to Edge 1, Edge 2, etc. and is updated using the locally sensed data.
2. These edges send the local updated model to the cloud.
3. The server at cloud aggregates the local model sent by edges into the global model.
4. The server sends the global model to all the edges concerned.
5. The edges integrate the global model, into their local model.

The communications between the edges and the server are encoded in binary format describing the updates along with any meta-data. This way the patient data privacy is ensured in addition to the no-sharing data policy. Several open challenges still exist in the areas of adaptation of an appropriate neural learning architecture for FL, data partitions that are to be made vertical or horizontal and others. Despite such challenges federated learning along with data processing at edge reduces the data travel time in the network, which will result in reduced latency and power requirements. Federated learning will be focussed more in our future work comprising of AWS cloud services (Green Grass for Edge services) and Raspberry Pi boards as edge nodes.

3.3 E-Health+ Process Flow

The process flow of the E-Health+ system is summarised in Fig. 3, where based on patients' requests arising due to health needs or the alarm raised due to the anomalistic sensor data, the data pre-processor swiftly decides on the nature of the situation as to emergency or not and appropriately routes the aggregated data to edge intelligence subsystem or to the cloud for deep analytics. The edge intelligence subsystem is the Open API services running in the edge server. It categorises the request based on the severity, nature and complexity to push it either to the federated learning or the resource allocation module. Here the selection between the two is facilitated by the core services running in the server.

The communications from the edge server to these modules belong to the publish/ subscribe method where the server just pushes the requests down the queue, thereby preventing any time delay. Both the FL and RA modules pick the requests as it keeps checking for new messages from the server. Resource allocation framework is a mechanism envisioned to route and designate suitable edge nodes to process the data quickly to obtain insight. With FL, the data is held at the patients' device which is acted upon by the ML algorithm to get the appropriate course of action. And the result is pushed back to the server, which can route it to the user.

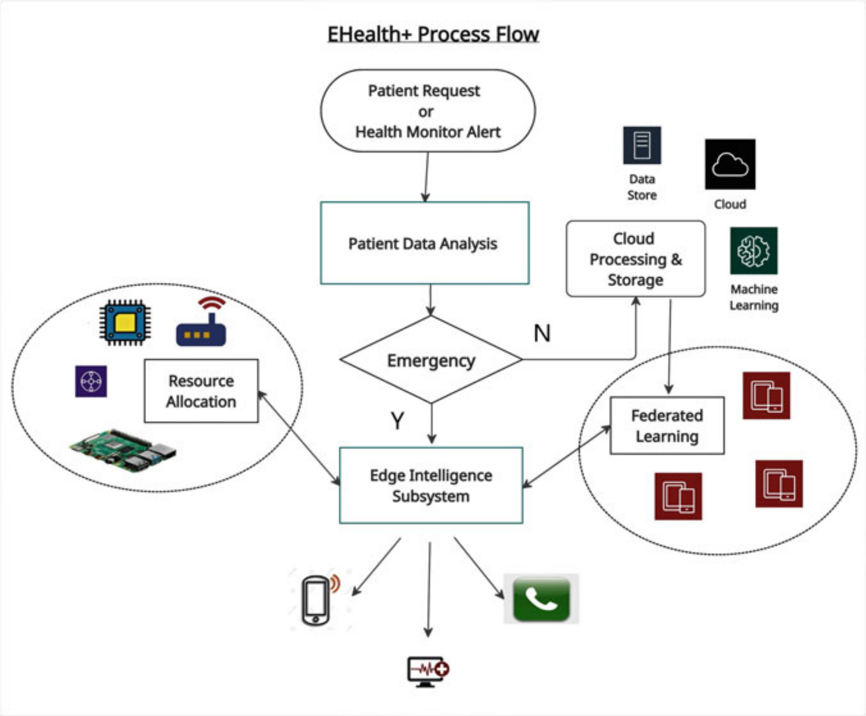


Fig. 3 Flow Diagram for EHealth System

3.4 *COVID-19 Data Analysis*

As part of our current work, we wanted to reaffirm the fact that by applying appropriate machine learning algorithm at edge devices, we can extract a wealth of hidden inferences from the sensed patient data. With this inference at hand, we can advise/ alert the connected patients accordingly. So, we started with the analysis of COVID-19 data from github [20], to gain appropriate insights about the nature of the pandemic, and come up with some predictions. We pre-processed and cleaned the data for inconsistencies using Python Scikit Library. Statistical summary, plots for COVID-19 cases, deaths worldwide, summary for cases in India have been generated which quickly summarise the COVID-19 scenario currently.

Starting with the worldwide COVID-19 summary (Fig. 4) clearly indicates the significant rise in the new confirmed, active and recovered cases, despite the marginal rise in death rates. This can be attributed to the improvement in the diagnosis, treatment and facilities like oxygen supplies ramped up by the governments.

COVID-19 cases summary worldwide since the year 2020 till date show an upward trend indicating the situation to be in the rise including second and third waves in different countries depending on the containment measures adapted. It can also be

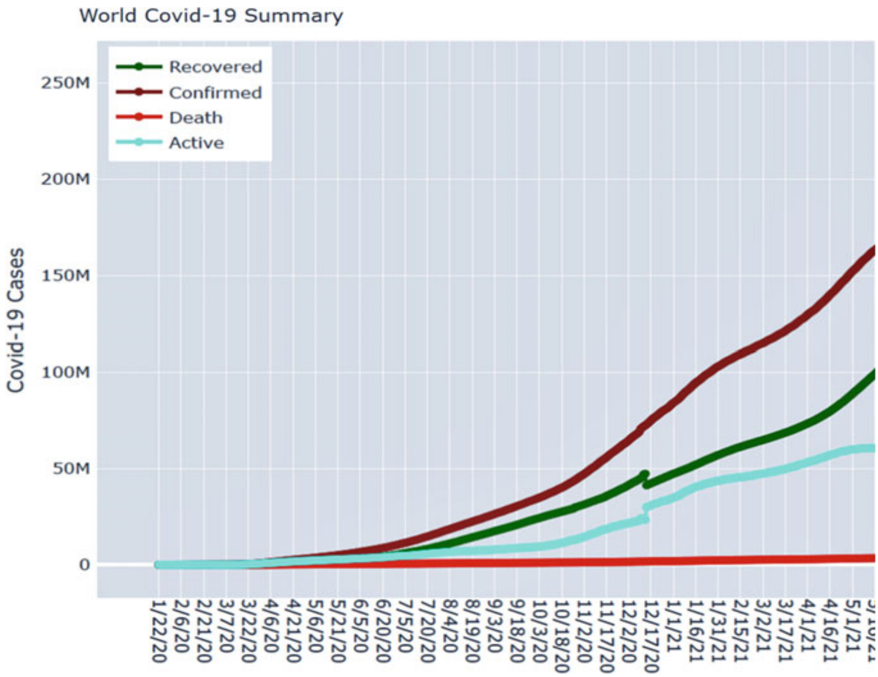


Fig. 4 COVID-19 summary as of Aug 2021

noted that, the death count is not rising in the same rate as that of COVID-19 case count indicating the recovery rate to be increasing, which is a promising trend in reduced mortality rate during second and third waves of this pandemic.

When comparing the current state of quick-paced vaccination progress (Fig. 5) in India and the rise in the new cases count, it clearly indicates the pandemic is not over. And the world still needs all the precautionary measures to be followed indicating the recovery rate to be increasing, which is a promising trend in reduced mortality rate during second and third waves of this pandemic.

It can also be noted that the vaccination drive in India has reached above 70 lakhs in September when compared with worldwide vaccination. And the dip in the vaccinations on certain dates corresponds to the non-availability of vaccines in different states.

The new COVID-19 cases count (Fig. 6) continues on a declining note with due credit for the aggressive vaccination drive adapted by the government.

Further, analysis based on the type of viral transmission presented in Tables 1 and 2 help us in quickly understanding the nature of the infection that was spreading due to the factors like population per square metre and the absence of face masks to block the viral entry into our system in different countries.

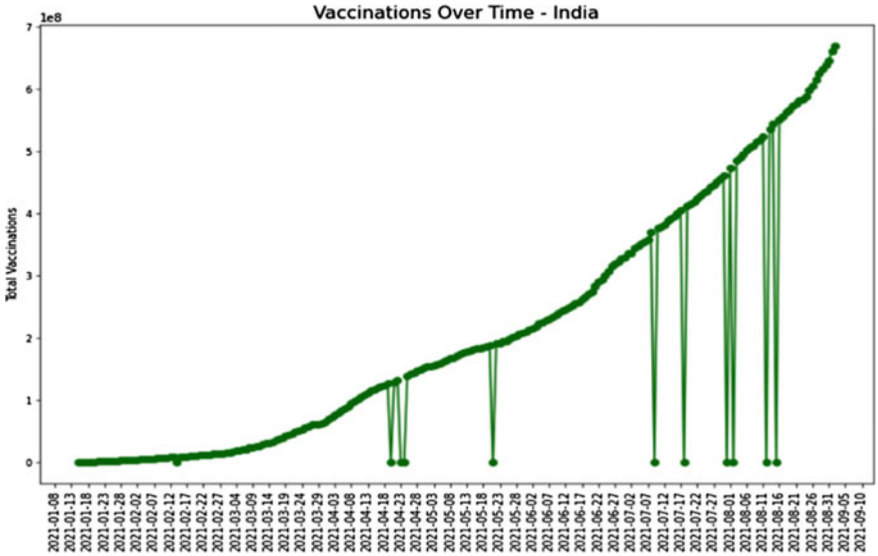


Fig. 5 Vaccination summary—India

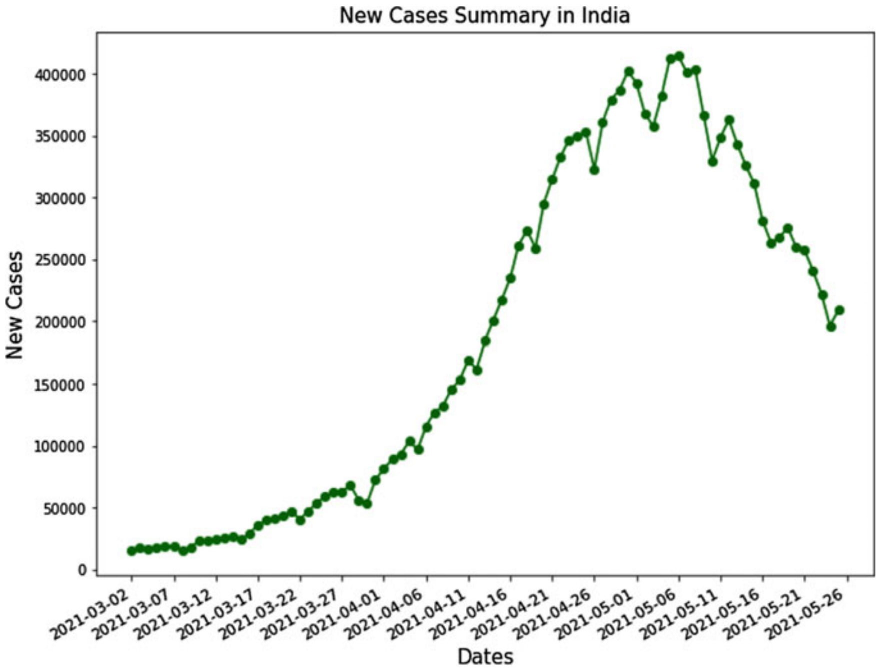


Fig. 6 New COVID-19 cases in India

Table 1 Countries with COVID-19 transmission type as "sporadic cases"

	Name	Cases—cumulative total	Deaths—cumulative total	Transmission classification
2	India	11,112,241	157,157	Clusters of cases
4	Russian Federation	4,257,650	86,455	Clusters of cases
8	Italy	2,925,265	97,699	Clusters of cases
24	Portugal	804,562	16,317	Clusters of cases
34	Morocco	483,654	8623	Clusters of cases
38	Japan	432,773	7887	Clusters of cases
44	Slovakia	308,083	7189	Clusters of cases
45	Malaysia	300,752	1130	Clusters of cases
48	Nepal	274,143	2774	Clusters of cases
50	Kazakhstan	262,725	3389	Clusters of cases
52	Bulgaria	247,038	10,191	Clusters of cases
55	Azerbaijan	234,537	3220	Clusters of cases
64	Slovenia	190,081	4111	Clusters of cases
66	Egypt	182,424	10,688	Clusters of cases
74	Myanmar	141,896	3199	Clusters of cases
79	Bahrain	122,395	449	Clusters of cases
81	Albania	107,167	1796	Clusters of cases

3.5 Results and Discussion

In the steady state of improvements seen with respect to COVID-19 cases, we have taken the active and recovered cases for comparison (Fig. 7) since the onset of the pandemic for different countries. And it can be clearly understood that the number of active cases is higher in India, despite the fact that the recovered cases are reported to

Table 2 Countries with COVID-19 transmission type as "Cluster of Cases"

	Name	Cases—cumulative total	Deaths—cumulative total	Transmission classification
41	Saudi Arabia	377,383	6494	Sporadic cases
97	Singapore	59,936	29	Sporadic cases
124	French Polynesia	18,387	139	Sporadic cases
153	Djibouti	6066	63	Sporadic cases
174	Liechtenstein	2642	52	Sporadic cases
183	Monaco	1953	24	Sporadic cases
190	Cambodia	820	0	Sporadic cases
193	Bermuda	705	12	Sporadic cases
194	Faroe Islands	658	1	Sporadic cases
196	Mauritius	610	10	Sporadic cases
199	Cayman Islands	438	2	Sporadic cases
201	Brunei Darussalam	186	3	Sporadic cases
203	Grenada	148	1	Sporadic cases
206	Timor-Leste	113	0	Sporadic cases
207	Fiji	59	2	Sporadic cases
208	New Caledonia	58	0	Sporadic cases

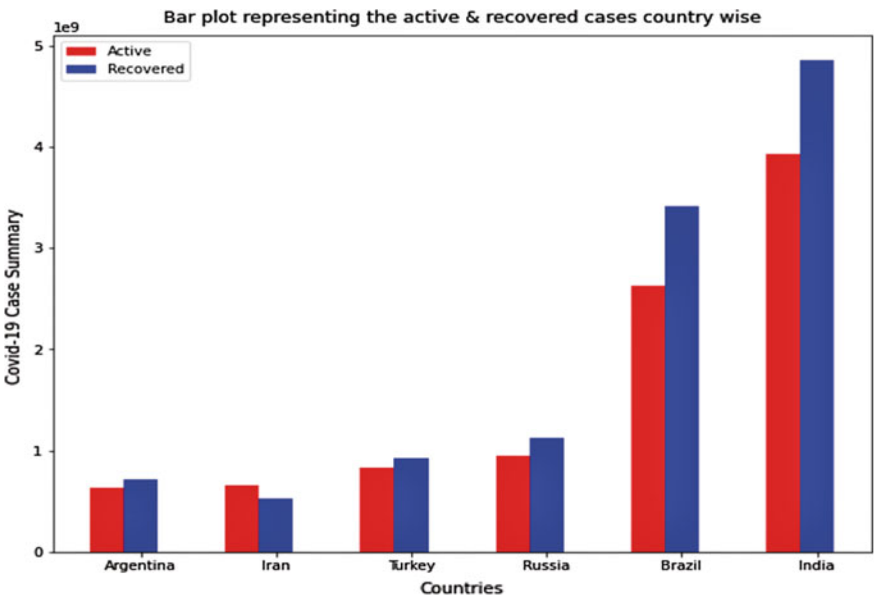


Fig. 7 Summary of active and recovered cases

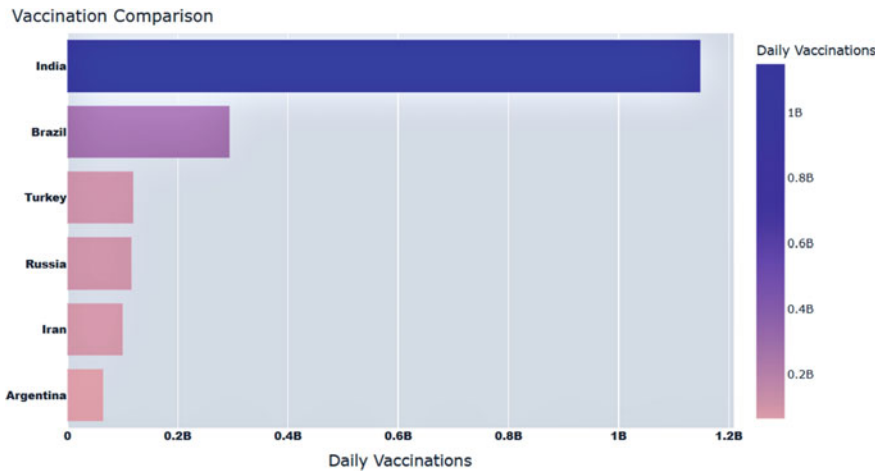


Fig. 8 Daily vaccinations summary for countries

be higher than the active cases. Also, it is to be noted that India ranks as the topmost level (Fig. 8) in terms of daily vaccinations when compared with the other countries considered.

Based on the exploratory data analysis carried out with the COVID-19 data, we could gain certain insights on the pandemic and try to look for unconventional ways to address the health issues faced by patients around the world. COVID-19 along with other emerging global epidemics, clearly indicate, healthcare solutions need to be innovative and time sensitive in order to save lives. In this aspect, our proposed system can provide a latent, remote health assistance anytime, anywhere.

The novelty of our proposition lies in the combinatorial deployment of using AI through federated learning and effective resource allocation framework that reduces latency. Although results in the above two significant research areas have shown to reduce latency in different domains, our objective is to prove that by integrating these two areas in the healthcare domain, a secure remote reliable and time bound health assistance can be provided for patients through our proposed work.

4 Conclusions and Future Enhancements

In this chapter, we proposed E-Health + system which is a contactless digital healthcare solution that can provide automated, personalised healthcare assistance based on the current health conditions of patients in pandemic situations or otherwise. The proposed system is to be evaluated with verifiable performance metrics with respect to latency and edge computing capabilities. We also performed a detailed exploratory data analysis on COVID-19, which helped in understanding the role artificial intelligence in understanding the vast amount of data generated by sensor devices to derive useful insights. Similar to these machine learning techniques used in analysis, appropriate learning models can be developed and deployed in the edge devices which is the next step in our work. This can help in achieving our objective to reduce latency and provide contactless health care.

References

1. Ndiaye M, Stephen S, Obi O, Adnan M, Abu-Mahfouz Hancke GP, Kurien AM, Djouani K (2020) IoT in the Wake of COVID: a survey on contributions, challenges and evolution. IEEE Access 8:186821–186839. https://ieeexplore.ieee.org/document/9220109
2. Gupta D, Bhatt S, Gupta M, Tosun AS (2021) Future smart connected communities to fight COVID-19 outbreak. Internet Things 13. https://doi.org/10.1016/j.iot.2020.100342
3. Sanyaolu A, Okorie C, Marinkovic A, Patidar R, Younis K, Desai P, Hosein Z, Padda I, Mangat J, Altaf M (2020) Comorbidity and its Impact on Patients with COVID-19. SN Compr Clin Med 1–8
4. Report by CDC, Centre for Disease Control and Prevention: Evidence used to update the list of underlying medica conditions that increase a person's risk of severe illness from COVID-19 May 2021 (Updated). https://www.cdc.gov/coronavirus/2019-ncov/hcp/clinical-care/underlyingconditions.html
5. Chen Y, Yang D, Cheng B, Chen J, Peng A, Yang C, Liu C, Xiong M, Deng A, Zhang Y, Zheng L, Huang K (2020) Clinical characteristics and outcomes of patients with diabetes and COVID-19 in association with glucose lowering medication. Diabetes Care 43. https://doi.org/10.2337/dc20-0660
6. Zhu L, She Z, Cheng X, Qin J, Zhang XJ, Cai J et al (2020) Association of blood glucose control and outcomes in patients with COVID-19 and pre-existing type 2 diabetes. Cell Metab 31:1068–1077. https://doi.org/10.1016/j.cmet.2020.04.021
7. Bode B, Garrett V, Messler J, McFarland R, Crowe J, Booth R, Klonoff DC (2020) Glycemic characteristics and clinical outcomes of COVID-19 patients hospitalized in the United States. J Diabetes Sci Technol 14(4):813–821. https://doi.org/10.1177/1932296820924469
8. Gottlieb M, Sansom S, Frankenberger C, Ward E, Hota MPH (2020) Clinical course and factors associated with hospitalization and critical illness among COVID-19 patients in Chicago, Illinois. Acad Emerg Med 27:963–973
9. Barron E, Bakhai C, Kar P, Weaver A, Bradley D, Ismail H (2020) Associations of type 1 and type 2 diabetes with COVID-19 related mortality in England: a whole-population study. Lancet Diabetes Endocrinol 8:813–822. https://doi.org/10.1016/S2213-8587(20)30272-2
10. An C, Lim H, Kim DW, Chang JH, Choi YJ, Kim SW (2020) Machine learning prediction for mortality of patients diagnosed with COVID-19: a nationwide. Sci Reports 10, Article number: 18716, Nature Report
11. Ning H, Li Y, Shi F, Yang LT (2020) Heterogeneous edge computing open platforms and tools for internet of things. Futur Gener Comput Syst 106:67–76
12. Gómez-Carmona O, Casado-Mansilla Kraemer FA, López-de-Ipiña D, García-Zubia J (2020) Exploring the computational cost of machine learning at the edge for human-centric Internet of Things. Futur Gener Comput Syst 112:670–683
13. Rieke N, Hancox J, Li W, Milletarì F, Roth HR, Albarqouni S, Bakas S, Galtier MN, Landman BA, Hein KM, Ourselin S, Sheller M, Summers RM, Trask A, Xu D, Baust M, Cardoso MJ (2020) The future of digital health with federated learning. Digital Med 3. https://doi.org/10.1038/s41746-020-00323-1
14. Pathinarupothi RK, Durga P, Rangan ES (2019) IoT-based smart edge for global health: remote monitoring with severity detection and alerts transmission. IEEE Internet Things J 6:2449-2462

15. Wang J, Wang L (2021) A computing resource allocation optimization strategy for massive internet of health things devices considering privacy protection in cloud edge computing environment. J Grid Comput 19:17
16. Lim WYB, Luong NC, Hoang DT, Jiao Y, Liang YC, Yang Q, Niyato D, Miao C (2020) Federated learning in mobile edge networks: a comprehensive survey. IEEE Commun Surv Tutorials 22(3), THIRD QUARTER
17. Firouzi R, Rahmani R, Kanter T (2021) Federated learning for distributed reasoning on edge computing. Networks and Technologies (ANT), Elsevier
18. Cooke RA, Fahmy SA (2020) Quantifying the latency benefits of near-edge and in-network FPGA acceleration. In: EdgeSys '20. ACM. https://doi.org/10.1145/3378679.3394534
19. Imakura A, Inaba H, Okada Y, Sakurai T (2021) Interpretable collaborative data analysis on distributed data. Expert Syst Appl 177
20. Github Dataset link. https://github.com/owid/covid19data/tree/master/public

Diabetes Disease Diagnosis Using Machine Learning Approach

Sonali Goyal, Neera Batra, and Kritika Chhabra

Abstract Diabetes is a condition in which blood glucose, called as blood sugar, is high in an abnormal way. If the prediction of disease is possible at an early stage, then the risk factors associated with diabetes can be considerably lower in severity. The main problem and highly challenging task are to predict diabetes accurately, and the reason of this challenge is the diabetes dataset's insufficient number of labels data and the existence of outliers. This research paper proposes a strong framework to predict the disease with the help of different types of machine learning (ML) algorithms: K-nearest neighbor (KNN), support vector machine (SVM), decision trees (DTs), Naive Bayes (NB), and logistic regression (LR). For implementation, a dataset has been taken from a PIMA database consisting patient's health record, and these five machine learning techniques are applied to that dataset. A comparison between all the algorithms is presented in this paper. The motive of the paper is to provide assistance to doctors with their practitioners for the early prediction of diabetes using ML algorithms.

Keywords Data mining · Predictive analysis · Machine learning algorithms · Healthcare · Diabetes

1 Introduction

Diabetes is a well-known word in today's globe, and it poses significant issues in both countries: developed and developing [1]. The pancreas produces the insulin hormone, which permits glucose to flow from food into the bloodstream. This problem, called as diabetes, occurs by a lack of hormone generated by pancreatic dysfunction, which can cause number of diseases: coma, retinal failure, joint failure, peripheral

S. Goyal (✉) · N. Batra
Computer Science & Engineering Department, MMEC, Maharishi Markandeshwar (Deemed to Be University), Mullana, India
e-mail: Sonaliguglani21@gmail.com

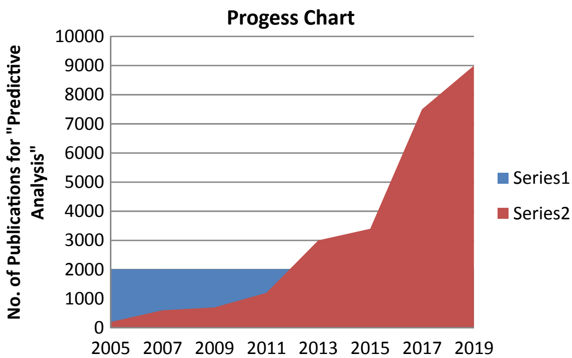
K. Chhabra
Ryerson University, Toronto, Canada

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_19

vascular diseases, cardiovascular dysfunction, cerebral vascular dysfunction, ulcers, sexual dysfunction, and pathogenic immune effects [2]. T1D (Type-1) diabetes is a metabolic illness characterized by high-blood glucose (BG) symptoms caused by inadequate endogenous insulin synthesis. In the Type-1 diabetes (T1D) research community, accurately predicting the levels of blood glucose has long been a type of challenge. Given the substantial heterogeneity in blood glucose among people, the best known theoretical compartment models frequently necessitate the identification for the model parameters that may be patient dependent [3]. Furthermore, an actual variation within an individual with different times of the day can create modeling becomes increasingly challenging [4]. Type 2 diabetes, on the other hand, is the most prevalent category, where the body is unable to effectively use the insulin produced [2]. In persons with diabetes, chronic hyperglycemia raises the risk of microvascular damage, which can lead to neuropathy, retinopathy, and nephropathy. As a result, diabetes is the primary cause of blindness and visual impairment in people in developed countries [2], accounting for more than one million lower limb amputations each year. Diabetics are also at an increased risk of macrovascular problems, since they are number of times more likely than non-diabetics to develop cardiovascular disease (CVD). If calculation is done for diabetes among adults category which are over 18 years old, it has climbed from 4.7 to 8.9% between 1980 and 2014, according to research, and is quickly increasing in countries [1]. According to statistics from 2017, 460 million people globally had diabetes, with that number expected to rise to 697 million by 2046 [1]. Another statistical study [1] demonstrates the severity of diabetes, reporting that nearly half a billion people globally have diabetes, with the numbers expected to rise to 26% and 52% in 2031 and 2046, respectively. Although there is no long-lived treatment for this disease, it can be managed as well as prevented if and only if an accurate early diagnosis is made. An analysis has been done on how much increase is there in publications with respect to year from 2005 to 2019 and it is shown in Fig. 1.

In last some years, lots of strategies were proposed and posted for the prediction of this diabetes disease by using number of algorithms: logistic regression, Naive Bayes (NB), support vector machine (SVM), K-nearest neighbor (KNN), and decision tree

Fig. 1 Number of publications w.r.t year



(DT). Recent traits in ML have enhanced the functionality of computer systems so that they can easily identify and label the given images, identification, and translation of speech, to play video games that includes skills and better IQ, disease prediction, and finally, it helps in taking decisions over data. For all these applications of ML, one of the main tasks is training, i.e., to train a computer system in a way that they can perform like humans or may be better than a human [5]. A number of supervised learning algorithms are there which can be used for training purpose with data labeling and after that, testing data will be used for the evaluation purpose. The remaining part of paper is organized as follows. In Sect. 2, it describes the literature review. Sections 3 and 4 provide the dataset used, methodology, and ML algorithms. Finally, results are defined in the Sect. 5, and future work is concluded in Sect. 6.

2 Literature Review

In last some years, there are number of researchers who worked in the area of data mining as well as predictive analysis mainly in healthcare industry with other areas for the prediction of the future challenges. There are number of data mining sources for the collection of data, data analysis, and prediction analysis. This paper focuses on machine learning algorithms for predictive analysis. There are number of researchers who worked for this area in healthcare industry. For this, some papers are mentioned here. In [6], Karik et al. use recent advances for the data analysis and its implementation in the healthcare area. This paper measures the diabetes disease while considering number of constraints and it provides an efficient platform for predictive analysis. A new visualization-based method is proposed in [7]. The aim of this paper is to find predominant features to predict whether tumor is benign or malignant. Exploratory data analysis technique is used for this prediction. Histograms are used in this paper to display a bivariate analysis of standard error, mean with worst mean providing the actual differences for every attribute. Nahla H. Barakat and Andrew P. Bradley define diabetes disease which is a public health issue worldwide. This paper [2] uses a number of machine learning algorithms for the diagnosis, prognosis as well as management of diabetes. This paper uses a hybrid system for the diabetes diagnosis in which SVM is used for diabetes diagnosis and prediction with rule-based component which is used for providing comprehensibility. Results show that this hybrid model is of high quality in terms of accuracy for diagnosis and prediction. In [8], Protima et al. presented a complete survey of detection for four most severe brain diseases like Alzheimer's disease (AD), epilepsy, and Parkinson's disease using ML and deep learning approaches. A review is on the basis of 147 recent papers on category of types of brain diseases using ML and DL approaches with twenty two datasets. A review of number of feature extraction methods is used for brain disease diagnoses, and a summary is finally designed for the major issues occurring in these areas (ML and DL). With the help of this study, it is easy to find the most accurate way to detect different brain diseases in betterment of future. In paper [9], a comparison is made between numbers of machine learning techniques to predict the advanced liver fibrosis present in chronic hepatitis C patients. A number of models were developed in this paper using particle swarm optimization, genetic algorithm, decision tree, and multi-linear regression. There are some parameters (sensitivity, specificity, criteria point, PPV, NPV, correlation coefficients, ROC, and accuracy) used in prediction of the advanced fibrosis by using some models (PSO, GA, MReg, ADT, and ADT*). Highest accuracy is achieved by proposed model using ADT* algorithm.

3 Dataset Used

In this proposed work, PIMA dataset is used; it is downloaded easily from the UCI repository. This dataset is having 770 patient records of Indian women along with 9 attributes. In Table 1, all the attributes used in this paper are mentioned and in Table 2, and there basic data statistics is defined. All the female patients are there in this dataset who belongs to Pima Indian heritage. The aim of this work is to predict diabetes disease: if a person has diabetes or not on the basis of medical measurements of the patient.

Table 1 Attributes of dataset

Attributes	Explanation
No_pregnancy	Number of pregnancies
Glu_conc	Plasma glucose concentration
Bp_dias	Diastolic bp value
Insulin	Insulin
BMI	Body mass index
Age	Age
Diabetes	Diabetes (Yes/No)

Table 2 Attributes with dataset statistics

Attributes	Count	Mean	Min	Max	Std
No_pregnancy	750	2.74	0.00	16	2.26
Glu_conc	750	119.79	0.00	198	30.87
Bp_dias	750	59.0	0.00	121	18.25
Insulin	750	78.69	0.00	845	114.1
BMI	750	30.89	0.00	66.0	6.78
Age	750	32.14	20.0	80.90	10.66

4 Methodology

Data mining, it is a process of finding out the data and fetching the appropriate information from large datasets by using number of data mining techniques like machine learning, artificial intelligence, and database systems. From these techniques, number of techniques of machine learning is used in this paper for diabetes prediction. This area provides a number of computer algorithms which helps to convert the raw data into meaningful information by the computer systems. Machine learning algorithms are basically of three types which are shown in Fig. 2.

Supervised learning means a technique where machine is trained using data or information which is well labeled. It is mostly used for classification and regression problems [10]. For e.g., to find email is spam or not, weather forecasting, stock analysis, etc. Unsupervised learning means machine is trained by using unlabeled data, and in this, it allows the machine to act accordingly on that information without any supervision. It includes association and clustering problems. For e.g., sales functions, word associations, etc. Reinforcement learning means in which an agent is put into an environment, and his task is to behave in that environment with the help of certain actions and by observing the rewards, it is getting from those actions. For e.g., self-driving cars [11]. This research paper evaluates the accuracy of five ML algorithms for the analysis of prediction for diabetes disease in healthcare so supervised learning is used in this work. The basic model that how supervised learning algorithm works is shown in Fig. 3.

Figure 4 shows an example of supervised category of learning classification technique. In supervised category, machine is actually fed with labeled data for the training purpose, where machine learns about each type of dataset. When training

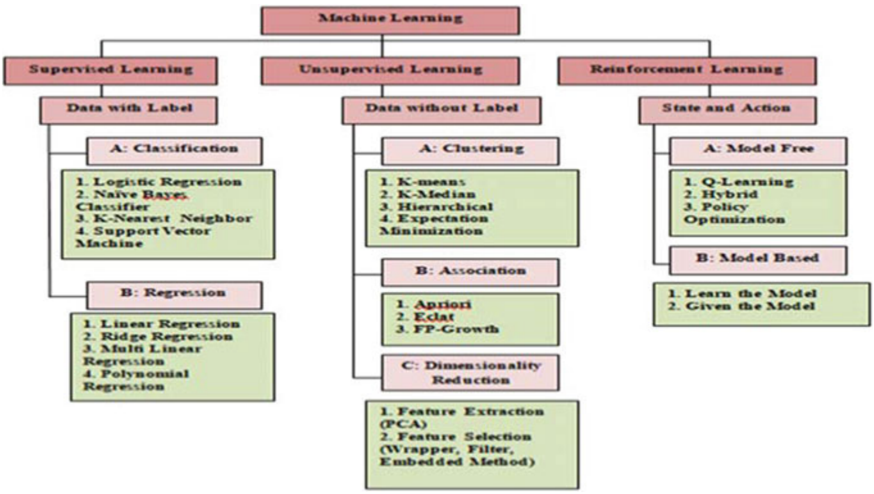


Fig. 2 Classification of machine learning

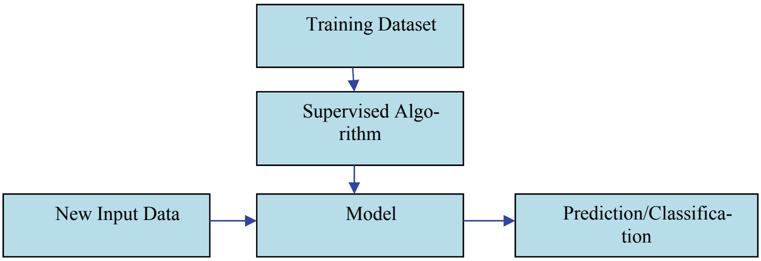


Fig. 3 Supervised learning model

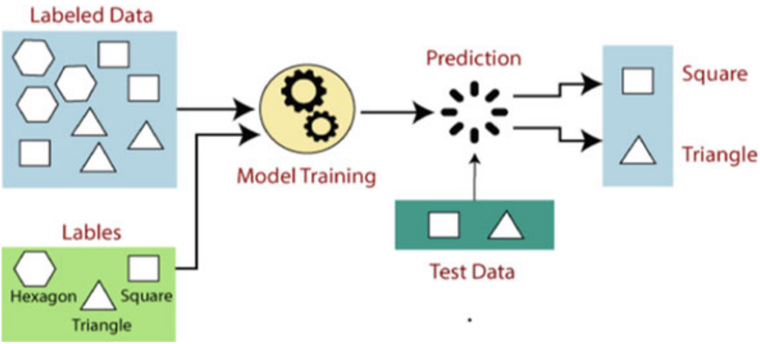


Fig. 4 Example of supervised learning model

process completes, the machine is now tested on the basis of test data, and finally, output is predicted [3]. This research paper uses five machine learning algorithms for the prediction of diabetes. These five algorithms are support vector machine (SVM), K-nearest neighbors (KNNs), logistic regression (LR), decision tree (DT), and Naive Bayes (NB).

5 Discussion Results

In this paper, five ML algorithms were taken for the experiment, and the name of these algorithms are support vector machine, K-nearest neighbor, logistic regression, decision tree, and Naïve Bayes. PIMA dataset is used by all these algorithms. This dataset is divided into two parts: one for training, other is for testing, and the ratio of training and testing is 70:30. These five algorithms are applied on this dataset, and results are calculated. The main attribute used for the evaluation of these algorithms is accuracy prediction. Accuracy means to calculate the overall success rate of an algorithm. The formula for calculating accuracy is

$$\text{Accuracy} = (\text{TP} + \text{TN})/(\text{P} + \text{N})$$

Here, TP means all predicted true positive, TN means true negative, and both these factors are divided by all positive and negative. There is a table also which is showing all the possible cases predicted by these five algorithms, i.e., true positive (TP), true negative (TN), false positive (FP), false negative (FN). In this paper, true positive means actual and predicted diabetes. False negative means actual diabetes but predicted to not diabetes. False positive means predicted diabetes but not actual diabetes. True negative means not actual diabetes and not predicted diabetes. From this Table 3, it is shown the indication of if patient will have diabetes or not.

Figure shows the results of accuracy by using five ML algorithms where support vector machine gives 76% accuracy, K-nearest neighbor gives 76% accuracy, logistic regression gives 73% accuracy, 70% accuracy is achieved by decision tree, and Naïve Bayes gives 73% accuracy. I means support vector machine and K-nearest neighbor provides highest rate of accuracy which is 76%. From these results, it can be concluded that SVM and KNN algorithm prove appropriated for the patients to predict the status of diabetes (Figs. 5 and 6).

Table 3 TP, FP, TN, FN predicted by algorithms

Algorithm	TP	FN	FP	TN
Support vector machine	35	17	35	139
K-nearest neighbor	40	19	22	135
Logistic regression	42	22	34	125
Decision tree	58	46	16	100
Naïve Bayes	50	30	26	115

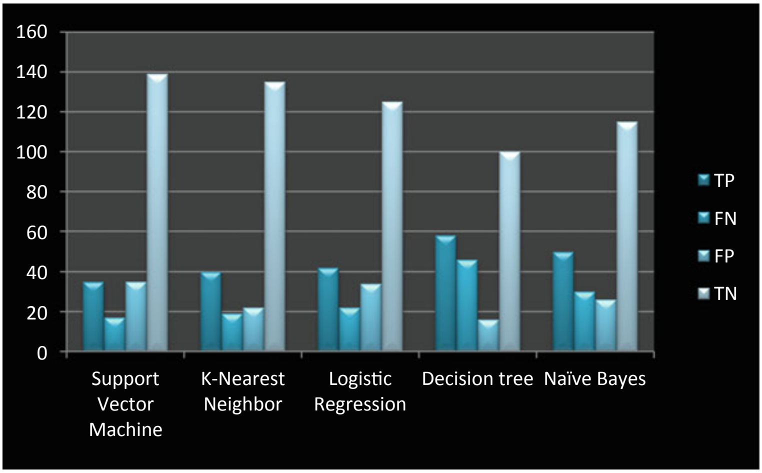


Fig. 5 TP, FP, TN, FN predicted by algorithms

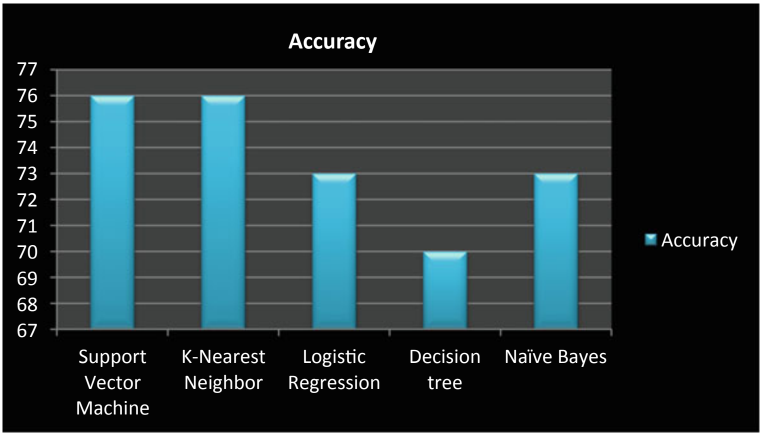


Fig. 6 Algorithms accuracy

6 Conclusion and Future Work

Analysis of prediction in healthcare industry has changed the outlook that how medical researchers achieve useful information from medical dataset and accordingly take the decisions. This paper uses five ML algorithms, i.e., SVM, DT, KNN, LR, and NB for the analysis of prediction about diabetes. It can be seen from the obtained results that highest accuracy of 77% is achieved by both SVM and KNN algorithm as compared to other three algorithms used in this paper for diabetes prediction. So, it can be concluded that these both algorithms, i.e., SVM and KNN are appropriated for diabetes prediction. There are some limitations of this research like dataset is small, and there may be some missing values sometimes. So, for higher accuracy and to recover these limitations, dataset can be enlarged with zero missing values. In future work, number of techniques can be employed in this paper for dealing with the problem of missing values.

References

1. Hasan K et al (2020) Diabetes prediction using ensembling of different machine learning classifiers. IIIE
2. Barakat NH (2010) Intelligible support vector machines for diagnosis of diabetes mellitus. IEEE, vol 14, no 4
3. Xie J, Wang X (2020) Benchmarking machine learning algorithms on blood glucose prediction for type I diabetes in comparison with classical time-series models. IEEE, vol 67, no 11
4. Dutta D, Paul D, Ghosh P (2018) Analyzing feature importances for diabetes prediction using machine learning. IEEE, pp 942–928
5. Mitushi, Varma S (2020) Diabetes prediction using machine learning techniques. IJERT 9(9)
6. Maheshwari R et al (2018) A machine learning based medical data analytics and visualization research platform. IEEE

7. Khan SA, Senthil Velan S (2020) Application of exploratory data analysis to generate inferences on the occurrence of breast cancer using a sample dataset. ICIEM
8. Khan P et al (2022) Machine learning and deep learning approaches for brain disease diagnosis: principles and recent advances. IEEE
9. Hashem S (2018) Comparison of machine learning approaches for prediction of advanced liver fibrosis in chronic hepatitis C patients. IEEE, vol 15, no 3
10. Vizhi K, Fash A (2020) Diabetes prediction using machine learning. IJAST 29(6):2842–2852
11. Davi C et al (2019) Severe dengue prognosis using human genome data and machine learning. IEEE, vol 66, no. 10

Efficient Virtual Machine Migration Algorithms for Data Centers in Cloud Computing

Krishan Tuli, Amanpreet Kaur, and Manisha Malhotra

Abstract As the technology is growing at a rapid pace, there is an increased demand for various cloud resources and resultant of which is the establishment of large number of cloud data centers (CDCs). A single cloud data center consumes large amount of energy, and eventually, it will lead to the higher operational cost and emission of carbon. To reduce the consumption of energy with better utilization of resources, different virtual machine (VM) and its consolidated approaches have been considered for the dynamic utilization of resources. In this paper, proposed enhanced artificial bee colony (PEA) has been proposed for better migration and placement of various VMs and physical machine (PM) dynamically. There are two distinct phases in this algorithm. Firstly, selection for the location of PM with access delay to the location where it needs to be migrated and secondly, reduction in number of VM migrations. Further, proposed approaches are compared to in terms of SLA-V, energy consumption, number of hosts shutdown, and resource utilization. Results show the gradual reduction in SLA-V by 20 and 31%, number of migrations by 16 and 25% and increase the resource utilization by 8%. There is a better improvement of 13% in energy consumption has been observed in the proposed method compared to others.

Keywords Cloud computing · VM placement · VM migration

1 Introduction

Cloud computing has made the revolution in the industry of technology by the provisioning in the on-demand service of resources to various users. These on-demand

K. Tuli (✉) · M. Malhotra
UIC, Chandigarh University, Mohali, India
e-mail: mca.krishantuli@gmail.com

M. Malhotra
e-mail: mmanishamalhotra@gmail.com

A. Kaur
Chitkara University Institute of Engineering & Technology, Chitkara University, Punjab, India

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_20

services have reduced the cost of many organizations as it is based on pay-as-you-go model. Many individuals and organizations can outsource various cloud services based on their demand of high-performance hardware, software applications, and licensed platform applications. Many organizations are dependent on various cloud services, and they prefer to outsource rather getting their private data centers for their various business needs. A cloud data centers need huge investment for its development, maintenance, and in management. As the demand of cloud services is increasing at a large scale, there is increased demand in establishing data centers as well. Therefore, it may cause a big problem of energy consumption by these cloud data centers. There is an increase of around 56% in energy consumption from 2005 to 2010 [1], and it has been reported that huge amount of energy consumption is only due to the mismanagement of resources at the various data centers [2]. As per the study [3], such servers are utilizing the energy consumption from 10 to 50% and this recorded for 5000 servers over a period of six months for monitoring the energy consumption behavior. There are several ways to reduce the energy consumption for the cloud data centers like switching the servers or power it off when it is not in use. Such use of energy consumption will result in the wastage of resources, and it will also increase the total cost of proprietorship. Such idle servers can consume up to 70% of the total consumption, and it will also have the impact on the environment due to higher carbon emission. As per the estimation by author [4], the carbon emission by CDC is 2% of global emission. There are two possible reasons for such wastage of power consumption in CDCs. Firstly, it is due to the inefficient usage of resources into the data centers and secondly due to the virtual machine (VM) migration and placement. Placement refers to the overloading and underloading VMs. When a host is using available resources but for completion of task, host needs more resources than such state is said to be overloaded. On the other hand, a host is known to be underloaded if the total number of available resources are less than or equal to a certain threshold level. Further, the overloaded hosts can lead to the problems of service level agreement violation (SLA-V), and cloud providers need to pay the penalty on the basis of SLA violations to the customer.

If one host is facing underloaded problem, then all the VMs need to be migrated to other hosts using live migration technique [5], and other underloaded hosts need to be powered off as per their need. This is the best way to reduce energy consumption as the servers which are now in working state need to be powered off. However, the consolidation of virtual machines may cause degradation of performance due to increase demand of resources. Further, if required resources will not be available to the user for processing, then it will lead to late response time and it may cause application failure. There are various approaches that have been proposed by many researchers to improve the utilization of resources with reduction in energy consumption. To reduce all such issues, we have proposed an approach for efficient VM migration approach keeping minimum SLA violation.

The proposed method is divided into four phases:

(a) First and second phase are used to detect the overloaded and underloaded hosts.

(b) Third phase is used for VM migration from various overloaded hosts to underloaded hosts.
(c) Last phase is used to detect other overloaded and underloaded hosts and make the final migration and placement of virtual machines.

Further, this approach is having two main features:

(a) To select the location of physical machine which has minimum delay compared to other PMs where the VM needs to be placed.
(b) To get the minimum number of migrations.

Resultant of this is the total number of migrations are reduced and hosts that are underloaded need to be powered off, which leads to reduction in energy consumption.

2 Related Work

The last decade had witnessed the scientific community being focused toward achieving energy efficient resource utilization for cloud environment. The virtualization technology has been the one of the most important aspect to achieve this goal. Therefore, many researchers had also been taking advantage of concept of optimization in order to achieve efficient resource allocation with minimal CPU and power consumption [5].

To address the VM migration issues, a variety of approaches have been proposed. Static resource scheduling algorithms including round-robin (RR), weighted RR, and destination and source hashing scheduling are among them. Dynamic methods, such as bin packing (BP), are another choice. Bin packing approximation algorithms are first fit algorithm (FF), best fit algorithm (BF), first fit decreasing algorithm (FFD), best fit decreasing algorithm (BFD), and modified BFD (MBFD).

A BP-based heuristic is used to investigate a concrete method for controlling the number of VMs migrated [6]. The approach prevents the migration of VMs with consistent workloads, which decreases the number of migrations. Beloglazov et al. propose a threshold-based VM migration optimization after a first fit BP process [7]. A multi-dimensional space partition model is used to coordinate different resource loads [8]. The resource space was divided into three categories by the authors: acceptance, protection, and forbidden domain. From the former to the latter, VM takes precedence. The given solution will effectively reduce the number of physical machines (PMs) required by making maximum use of the complementary resource requirement. Song et al. (2013) present an adaptive approximation algorithm based on the migration of BP and VMs. VMs are divided into four categories, ranging from small to big, for the one dimension case, and then packed according to the categories. It has been established that the algorithm is approximate. Only the number of batch VM migrations can be reduced using the algorithm. However, it cannot guarantee the number of VMs that will be migrated [8].

A new formulation: multi-capacity stochastic BP problem is proposed, which is similar to but different from BP. The new one differs from multi-dimensional BP in that it believes that a part of one resource allocated to one VM is only used by that VM. Multi-capacity BP more realistically captures server resource characteristics. To reduce the amount of stochastic resources used, a two-stage heuristic algorithm is provided [9]. Heuristics based on BP are often widely used to save resources by reducing the number of services needed.

A significant number of evolutionary methods have been explored to address the optimization problems of cloud data centers, according to the literature review.

3 Proposed Enhanced Artificial Bee Colony (PEA)

For VM migration, PEA works on four different phases as shown in Fig. 2. The following are the various stages in the form of flowchart that shows how they are employed for VM migration. First of all, underloaded and overloaded hosts need to be detected across all the cloud centers. In this phase, the VMs from overloaded PMs need to be migrated to the underloaded PMs according to its nearby location. The reason to get the nearby locations is that it gives minimum delay to the cloud user. This process follows till all the overloaded PMs goes to its normal state. A list of pending overloaded and underloaded PMs is returned to the final migration algorithm. Moreover, all the PM list is further divided into two halves. The VMs that are there in first half list (FHL) will be migrated to the second half list (SHL). When all the PMs from SHL become normal, the VMs from pending PMs will be migrated to new or other PMs. The detailed phase is discussed in the next subsections (Fig. 1).

3.1 Detection of Overloaded Hosts

When an VM is allocated in the first phase, the algorithm analyzes the overloaded PMs periodically across all the data centers. This algorithm detects the overloaded hosts that are running in data centers and place all these overloaded hosts in the overloaded host list, and further, this list will be passed to migration and placement algorithm. Algorithm 1 is used to perform the detection of overloaded hosts in all the data centers.

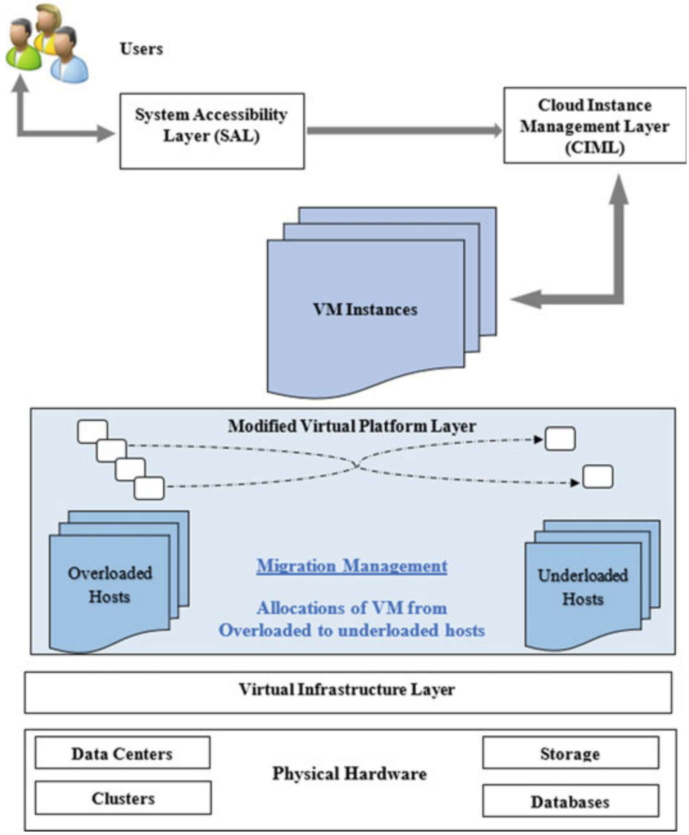


Fig. 1 Migration management SIM-cumulus architecture

Algorithm 1 Overloaded Host Detection

Input: PM $_{List}$ = List of all the PMs in the Data Centers
Output: List of Overloaded PMs.

Initialize pmUtilization = 0
Initialize pmTotalCapacity = 0
Initialize overutilizedList = 0
for all PM in PM $_{List}$ do
    pmTotalCapacity.get.TotalMips[PM]
    pmTotalCapacity.get.TotalMips[PM]
    if (pmUtilization ≥ pmTotalCapacity) then
        Add PM to overloadedPMsList
    end if
end for
Return

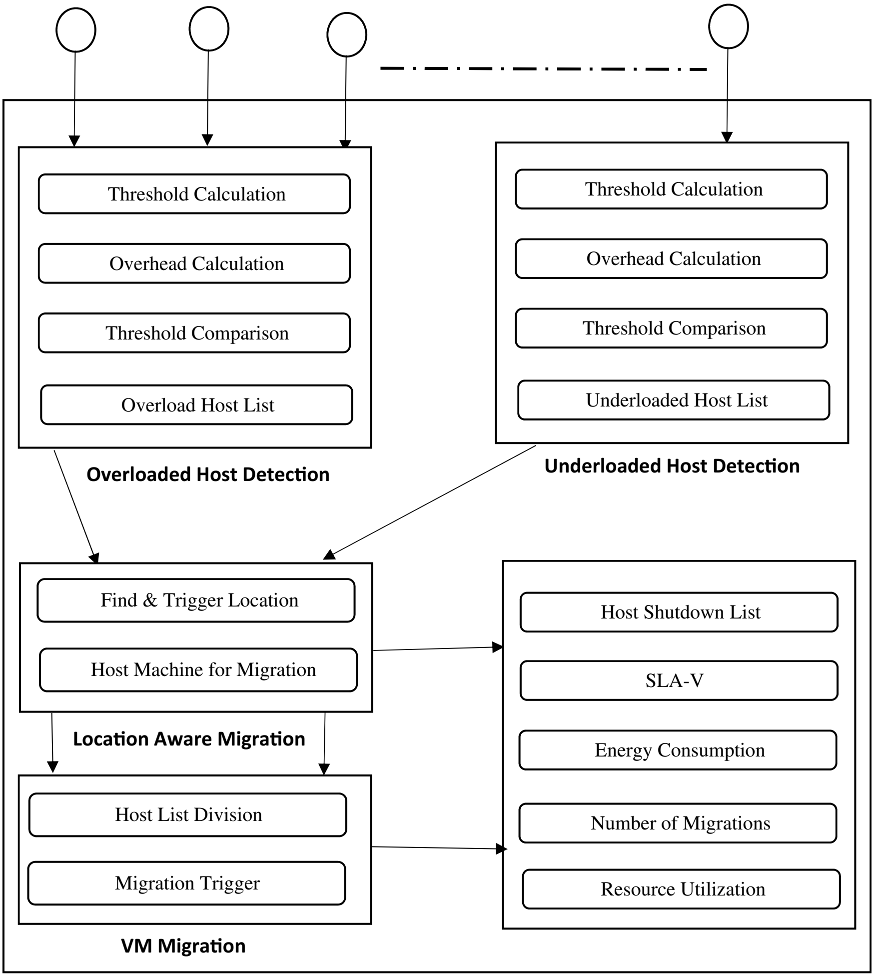


Fig. 2 VM migration flowchart

3.2 Detecting Underloaded Hosts

In the given phase, first of all, all the PMs need to be detected that are running at below threshold. Lower value of threshold is 35% of the total CPU. A complete list of underloaded hosts needs to be figure it out, and further, it is passed for the migration and placement of VMs. From the algorithms, once we have detected overloaded hosts, the next step is to get the underloaded hosts.

Algorithm 2 Underloaded Host Detection

Input: pmList = PM List i.e PM1, PM2, PM3….PMn
Output: underloadedPMsList

1: Initialize pmUtilization = 0
2: Initialize pmTotalCapacity = 0
3: Initialize lowerTheshold = 0
4: **for all** PM in pmList do
5: pmTotalCapacity = PM.get.TotalMips[PM]
6: pmUtilization = PM.get.Utilization[PM]
7: lowerThreshold = pmTotalCapacity × 35/100
8: if (pmUtilization ≤ lowerThreshold) then
9: Add PM to underloadedPMsList
10: **end if**
11: **end for**
12: **Return** underloadedPMList

In the above algorithm, host utilization is checked based on the lower threshold. PM total capacity is holding the MIPS of PMs. In the step 4 of the algorithm, all the PMs need to be analyze one by one in their data centers. Further, in step 7, it is calculating the lower threshold from the total capacity of the PMs and their lower value of threshold which is 35%. After evaluation of all the PMs from the data centers, complete list of underloaded hosts is prepared. This list is further send to the migration and placement algorithm for finally migration and placement of virtual machines.

3.3 Migration and Placement of VM

This is the next phase in which VMs of underloaded hosts from Algorithm 2 need to be migrated to the other underloaded hosts based on user's location. Data centers which are near to VMs need to be migrated because they are have minimum delay. Once detection of underloaded and overloaded host is done, next step is to migrate the virtual machines. The migration will be done from overloaded hosts to underloaded hosts. The main focus is on the nearby PMs because they give minimum delay in providing the service to the other cloud hosts. The list of underloaded PMs needs to be sorted in descending order for VM migrations [10, 11]. First of all, the VMs will be placed from overloaded to underloaded PM. Then, in Algorithm 3, it analyzes the overloaded PM from the list of overloaded PM. Selected overloaded hosts need to be sorted in ascending order considering utilization of MIPS. In Algorithm 3, line 11, the size of VMs is compared with the available underloaded PMs. If capacity on underloaded PM is not capable to handle, then VM is placed and next PM is selected for the VM. At last, list of pending underloaded and overloaded PMs is finally returned to migration algorithm [12].

Algorithm 3 VM Migration & Placement

Input: underloadedPMsList and overoadedPMsList
Output: underloadedPMsList

```
Initialize undrloadedPMs.sortDec()
for all OPM in overloadedPMsList do
  RemOverLoadedHosts = overloadedPMsList − 1
  OPM.VM.sortAsc()
  for all VM in overloadedPMsList do
    for all UPM in underloadedPMsList do
      RemUndLoadedHosts = underloadedPMsList − OPM
      if (uderloadedPM.location == VM.location) then
        if (VM.size ≤ underloadedPM.size) then
        Add PM to underloadedPMsList
        PlaceVM(undrloadedPM, VM)
        addNormalList1(undrloadedPM, PM.ID)
      end if
    if ((underloadedPMlist == 0) && (RemOverLoadedHosts == 0)) then
      break;
    end if
  end for
end for
Return underloadedPMsList
```

3.4 Final Migration

This is the last phase of proposed algorithm. In this given phase, there are some pending overloaded and underloaded hosts. This migration is executed only when the underloaded and overloaded hosts list are finalized, and VMs are migrated from overloaded PMs to underloaded PMs. Final migration is done from the pending list of underloaded and overloaded host list. Algorithm 4 divides the underloaded PM list into two halves. First one is known as first half list (FHL), and second one is called second half list (SHL). The VMs in the FHL are migrated to SHL. In this way of migration, there are two benefits. Firstly, the number of migration is reduced and secondly, increases in host shutdown.

Algorithm 4 Final Migration

Input: underloadedPMsList
Output: FinalizedList

```
Initialize FirsthalfUnderPM
Initialize SecondhalfUnderPM
undrloadedPMs.sortDec()
for all FHPMs in FirsthalfUnderPM do
   for all SHPMs in Secondhal fUnderPM do
      for all VM in Secondhal fUnderPM do
        if (VM.size < FHPMs) then
           PlaceVM(FHPMs, VM)
           Break;
           if (FHPMs.utilization == 0) then
              Break;
           end if
        else
           Break;
        end if
      end for
   end for
end for
```

4 Results and Validations

To obtain the result of PEA approach, the algorithm is compared based on SLA violation, energy consumption, hosts shutdown, number of migrations, and resource utilizations. The illustration of the ordinal measures and the results are given by Table 1.

The proposed work shows significant improvement due to the selection of appropriate VMs on the appropriate host by utilizing the idol time of the VMs at the hosts. As a result, more power is preserved as shown in Table 3 when analyze over 10,000 VMs. The migration analysis shows that the average VM migrations observed for Nashaat et al. are 2802.18 which is 2123.18 using Masdari and Khezri. In comparison with these two works, the PEA achieved only 1637.41 migrations. This exhibits

Table 1 Ordinal measure

Maximum number of VMs	10,000
Minimum number of VMs	200
Supplied load	10^6MIPS
Total number of simulations per VM set	100
Maximum number of simulations	10,000 × 100
Implementation tool	Python
Tool set	Spyder
Platform	Anaconda

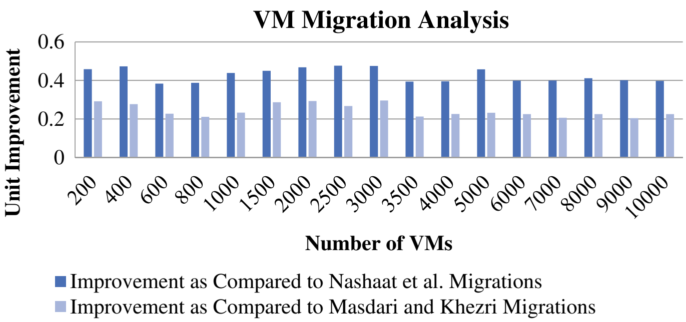


Fig. 3 Unit improvements for migrations

an average improvement of 1164.76 and 485.76 observed for PEA with respect to Nashaat et al. and Masdari and Khezri, respectively. The unit improvement with respect to increase in the number of VMs is further illustrated in Fig. 3. It is observed that the unit improvement in terms of VM migrations lies between 0.3838 and 0.4766 with an average unit improvement of 0.4275.

As the number of VMs is increased, there is considerable increase in the power consumption either using existing works or the PEA. However, it is observed that the average power consumption of PEA is only 286.62 in comparison with 512.20 using Nashaat et al. and 378.29 using Masdari and Khezri. In other words, this shows that the PEA exhibits 225.57 and 910.66 units less power consumption as compared to the Nashaat et al. and Masdari and Khezri, respectively. The unit wise improvement observed against two existing works is illustrated in Fig. 4 that shows that the unit improvement for power consumption ranges from 0.376 to 0.485 for Nashaat et al. and from 0.202 to 0.294 for Masdari and Khezri work. The average unit improvement demonstrated by PEA against Nashaat et al. and Masdari and Khezri is 0.435 and 0.243, respectively.

The quality of service delivered at the client side is determined by the number of SLA-V. The SLA-V analysis of the PEA shows that it exhibits least SLA-V with an

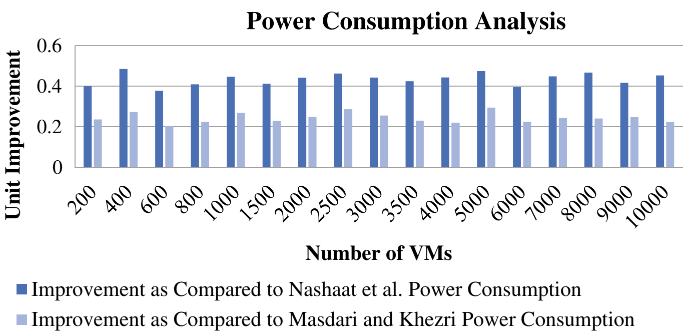


Fig. 4 Unit improvements for power consumption

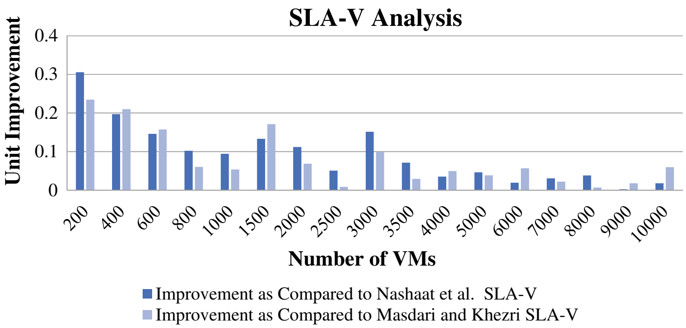


Fig. 5 Unit improvements for SLA-V

average value of 0.7011 followed by Masdari and Khezri of 0.759 and Nashaat et al. of 0.769. This shows that PEA exhibits 0.058 and 0.068 less SLA-V as compared to average SLA-V of Masdari and Khezri and Nashaat et al., respectively, when analyzed over 10,000 VMs.

The unit improvement in terms of SLA-V observed for PEA against each variation in the number of VMs used for the experimentation is shown in Fig. 5. It is observed that the unit improvement lies between 0.0025 and 0.3059 when compared against Nashaat et al. and between 0.0072 and 0.234 when compared against Masdari and Khezri. In other words, PEA exhibits an average unit improvement of 0.091 and 0.792 units against Nashaat et al. and Masdari and Khezri, respectively. The detailed simulation analysis in terms of three important criteria shows that the PEA successfully reduced the number of VM migrations even with increase in the number of VMs in each step. Further, it is observed that in comparison with the existing studies. PEA exhibits least power consumption and SLA-Vs, when analyzed using 10,000 VMs.

5 Conclusion and Future Works

Energy consumption and environmental sustainability of modern CDCs have become a major concern for cloud service providers (CSPs). Due to the increasing demands for cloud services, CSPs are interested in the realization of energy-efficient methods to significantly reduce energy consumption. In this paper, we present VM migration and placement approach that significantly improves the energy consumption and QoS provisioning in CDCs. The proposed approach is based on the principle of migrating VMs from overloaded hosts to underloaded hosts while keeping the location (of VMs/users) intact with less delay that ultimately leads to a reduction in energy consumption. The obtained results demonstrate that our approach reduces energy consumption by 13%, reduces SLA violation by 15%, the number of hosts shutdown by 10–13%, the number of migrations by 19%, and resource utilization by 20% as compared to the existing approaches.

References

1. Beloglazov A, Buyya R (2010) Energy efficient resource management in virtualized cloud data centers. In: 2010 10th IEEE/ACM international conference on cluster, cloud and grid computing. IEEE, pp 826–831
2. Le Sueur E, Heiser G (2010) Dynamic voltage and frequency scaling: the laws of diminishing returns. In: Proceedings of the 2010 international conference on Power aware computing and systems, pp 1–8
3. Masdari M, Khezri H (2020). Efficient VM migrations using forecasting techniques in cloud computing: a comprehensive review. Cluster Comput 1–30
4. Zhang F, Liu G, Fu X, Yahyapour R (2018) A survey on virtual machine migration: challenges, techniques, and open issues. IEEE Commun Surv Tutor 20(2):1206–1243
5. Zhang J, Huang H, Wang X (2016) Resource provision algorithms in cloud computing: a survey. J Netw Comput Appl 64:23–42
6. Ferreto TC, Netto MA, Calheiros RN, De Rose CA (2011) Server consolidation with migration control for virtualized data centers. Futur Gener Comput Syst 27(8):1027–1034
7. Beloglazov A, Buyya R (2012) Managing overloaded hosts for dynamic consolidation of virtual machines in cloud data centers under quality of service constraints. IEEE Trans Parallel Distrib Syst 24(7):1366–1379
8. Li X, Qian Z, Lu S, Wu J (2013) Energy efficient virtual machine placement algorithm with balanced and improved resource utilization in a data center. Math Comput Model 58(5–6):1222–1235
9. Yavari M, Rahbar AG, Fathi MH (2019) Temperature and energy-aware consolidation algorithms in cloud computing. J Cloud Comput 8(1):1–16
10. Zhang P, Zhou M, Wang X (2020) An intelligent optimization method for optimal virtual machine allocation in cloud data centers. IEEE Trans Autom Sci Eng 17(4):1725–1735
11. Arroba P, Moya JM, Ayala JL, Buyya R (2017) Dynamic Voltage and frequency scaling-aware dynamic consolidation of virtual machines for energy efficient cloud data centers. Concurr Comput Pract Exp 29(10):e4067
12. Nashaat H, Ashry N, Rizk R (2019) Smart elastic scheduling algorithm for virtual machine migration in cloud computing. J Supercomput 75(7):3842–3865

Lung Disease Detection Using Machine Learning Approach

Neera Batra, Sonali Goyal, and Kritika Chhabra

Abstract The objective of this work is to identify clinical factors that modulate the risk of progression to lung diseases such as asthma, chronic obstructive pulmonary disease (COPD), emphysema, lung cancer, bronchitis, and allergies among patients using data extracted with assistance from machine learning algorithms. In this work, we have gathered 250 instances along with 14 attributes. These information have been gathered from patients experiencing various lung illnesses alongside different indications. The lung illnesses trait contains two sorts of class which are 'Positive' and 'Negative.' 'Positive' implies that the individual has lung illness. The dataset has been trained using K-fold cross-validation technique. Four machine learning algorithms have been used for analysis which are logistic regression, random forest, KNN, and Bayesian networks.

Keywords Lung diseases · Logistic regression · Random forest · KNN · Bayes net · Machine learning algorithm

1 Introduction

In the world, respiratory disease is the major cause of morbidity and mortality. The most common main diagnosis among four common respiratory disorders, including COPD, was asthma in the Indian subgroup of the Asia-Pacific burden of respiratory diseases research. The most major contributors to the disease burden were lost productivity and drug costs. In India, secondary data analysis from the second phase of the Indian Human Development Survey revealed that asthma prevalence is

N. Batra (✉)
MMEC, M. M. (Deemed to be) University, Ambala, India
e-mail: batraneera1@gmail.com

S. Goyal
M. M. (Deemed to Be) University, Ambala, India

K. Chhabra
Ryerson University, Toronto, ON, Canada

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_21

increasing. Asthma prevalence was higher in impoverished states and among people who lived in homes that used solid fuels. Currently, the precision of lung disease plays an important role. Data mining helps in uncovering hidden patterns in medical data that is useful in diagnosing various ailments. Data mining enables health organizations to analyze and use data in a systematic way to uncover inefficiencies and best practices that enhance treatment and lower costs. Early diagnosis of lung disease is one of the most significant issues, and a lot of research is being done to develop a smart and accurate decision support system to assist physicians be more effective. As a result, we conducted a search using machine learning techniques to find the most relevant information. In this work, we investigated to determine the likelihood of lung disease occurrence using machine learning algorithms before it hits.

2 Related Work

Bharati [1] used using various classification algorithms for the lung cancer disease diagnosis. Nidhi S. Nadkarni and Prof. Sangam Borkar's paper [2] is primarily concerned with the classification of lung pictures as normal or pathological. The use of mathematical morphological operations allows for reliable lung segmentation and tumor detection. Three geometrical features area, perimeter, and eccentricity were given to the SVM classifier for classification. Moradi et al. [3] examined various approaches for distinguishing lung cancer nodules from non-nodules in 2019. In [4], R-CNN algorithm was used to detect the lung nodules. The fusion uses a deep CNN architecture with a dual path network (DPN) to classify and extract data. In this paper [5], the classification is performed and the results were evaluated with the performance comparison of various algorithms. This prediction system is useful for the doctors to take an appropriate decision based on patient's condition. In this paper, Sadiya et al. [6] used machine learning techniques for Tuberculosis and Pneumonia diagnosis so as to differentiate them at an early stage and prevent the misdiagnosis due to similar symptoms.

Ismail [7] aimed to detect the lung cancer at an early stage with more accuracy levels using various machine learning algorithms. Aykanat et al. [8] used two different types of machine learning algorithms: MFCC features in a support vector machine (SVM) and spectrogram pictures in a convolutional neural network (CNN). Khobragade et al. [9] used using chest radiographs and histogram equalization in image segmentation for image preprocessing, and feedforward neural network for classification purpose in order to detect lung disease. Park et al. [10] developed a neural network model for lung disease detection and the performance of each model was assessed using F1-scores.

3 Methodology

The disease prediction model based on machine learning algorithms is described in this section.

- **Data Collection and Preprocessing**

The data for this analysis has been collected from Centre for Integrative Medicine and Research (CIMR), Delhi. It consists of 250 instances and 14 attributes to analyze the lung disease in a patient. The preprocessing has been done by utilizing the two unsupervised filters. Replace missing values filter replaces all the missing qualities for ostensible and numerical traits utilizing the modes and means and the randomize filter replaces the missing data.

- **Data Training**

The K-fold cross-validation method is used to train the dataset.

- **Methodology**

In this phase, classification has been done using machine learning algorithms such as logistic regression, random forest, KNN, and Bayes Net. To diagnose the disease, some symptoms shown by patients such as coughing, wheezing, yellow fingers, fatigue, peer pressure, anxiety, allergy, smoking, alcohol consuming, and chest pain are used to which further helped in selection of features to be used for lung detection (Table 1).

- **Workflow**

The overall workflow of the entire analysis process is represented in Fig. 1 with the help of a flowchart.

The dataset comprising of 250 instances along with 14 attributes collected from Centre for Integrative Medicine and Research (CIMR), Delhi, and the data is preprocessed using feature selection option. It has been trained using K-fold cross-validation technique. To build the model, four machine learning algorithms are used, viz. logistic regression, Bayes, KNN, and random forest. The analysis has been performed by comparing the performances of the four algorithms.

- **Performance Parameters**

The following performance parameters are used for the analysis:

A. Kappa Statistics (KS): a measure of how closely the instances classified by the machine learning classifier matched the data labeled as ground truth.

$$k = \frac{R_O - R_e}{1 - R_e}$$

where $= R_O$the observed agreement.

$R_e =$ the expected agreement.

Table 1 Features list

Features	Range and subcategory
Age	(12–75 Years)
Sex	Male
	Female
Hemoglobin (Hb)	(6–14) gm/dl
White blood cell (WBC)	4.5 to 11.0 × 109/L
Platelet count (PC)	140 to 400 K/uL
Hematocrit (HCT)	(9 – 54)%
Neutrophils	(6 – 90)%
Erythrocyte sedimentation rate (ESR)	(1 – 160) mm
Lymphocytes	(1 – 48)%
Monocytes	(2 – 8)%
Eosinophil	(0 – 16)%
Serum-creatinine	(0.5 – 1.2) mg/dl
Serum-bilirubin	(0.3 – 1.2) mg/dl
Serum glutamic pyruvic transaminase (SGPT)	(7 – 56) u/l
Lung-disease-result	Negative: 30
	Positive: 220

B. Mean Absolute Error (MAE): It is used to measure the closeness of the prediction to the eventual outcomes

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - x|$$

where $n =$ the number of errors.
$|x_i - x|$ is the absolute error.

C. Root Mean Square Error (RMSE): The root mean square error is a measure of the differences between values predicted by a model and the values observed.

$$RMSE = \sqrt{\sum_{i=1}^{N}\frac{\left(x_i - x_i^{\wedge}\right)^2}{N}}$$

where i is a variable
$N =$ number of non-missing data points
$x_i =$ actual observations
$x_i^{\wedge} =$ estimated observations.

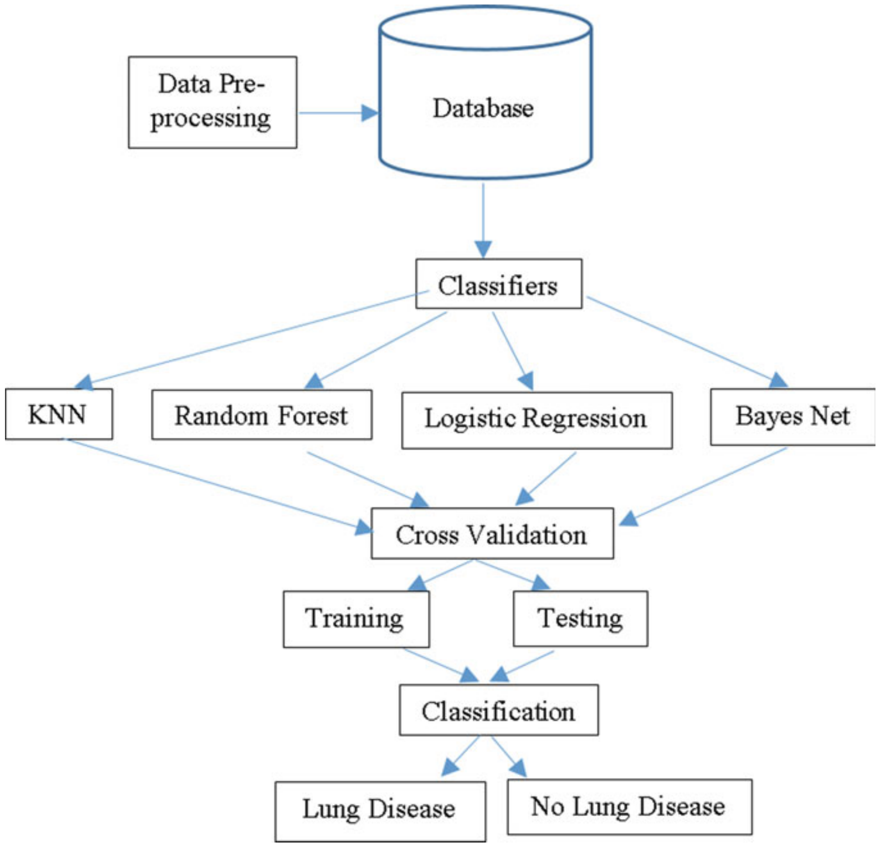


Fig. 1 Flowchart for proposed prediction model

D. Relative Absolute Error (RAE): The relative absolute error is expressed as a ratio, comparing a mean error (residual) to errors produced by a trivial or naive model. A reasonable model (one which produces results that are better than a trivial model) will result in a ratio of less than one.

$$RAE = \frac{\sum_{i=1}^{n} p_i - a_i}{\sum_{i=1}^{n} |\overline{a} - a_i|}$$

where p_i = predicted target.
a_i = actual target.

E. Root relative squared error: the average of the actual values
F. TR rate (sensitivity): TPR is the probability that an actual positive will test positive.

It is calculated as:

$$\frac{T_p}{T_p + F_n}$$

where T_p = True Positive.
F_n = False Negative.

G. FR Rate (Specificity): It is the probability that an actual negative will test negative.

It is calculated as

$$\frac{T_n}{T_n} + F_p$$

where T_n = True Negative.
F_p = False Positive.

H. Precision (PRE): It is the fraction of relevant instances among the retrieved instances (also called positive predictive value)

$$PRE = T_p/\left(T_p + F_p\right)$$

I. Recall: It is denoted as is the fraction of relevant instances that were retrieved.

$$R = T_p/\left(T_p + F_n\right)$$

J. F-Measure: A way to combine both precision and recall into a single measure that captures both properties, giving each the same weighting.

$$FM = (2 * PRE * R)/(PRE + R)$$

4 Results Analysis

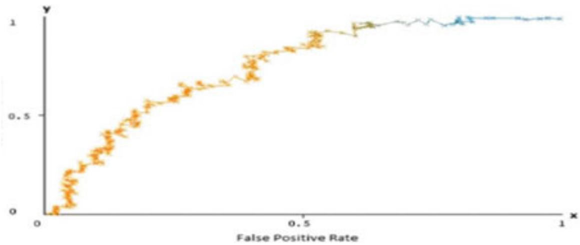
The results obtained from the model are summarized in the following table.

As it becomes clear from Table 2, Bayes Net shows poor performance with the highest percentage of error, i.e., 81.23% and random forest outperform all the other algorithms by showing the lowest proportion of these errors (70.12%) (Fig. 2).

The root relative-squared error shown by the Bayes net, RF, LR, and KNN is 97.45%, 90.23%, 92.12%, and 94.76%, respectively. Figure 3a shows the correct classification of instances, and Fig. 3b shows the incorrect classification of instances done by four algorithms.

Table 2 Results analysis

Evaluation Metrix	Logistic Regression (LR)	Random Forest (RF)	Bayes Net	KNN
Kappa statistic	0.371	0.497	0.3012	0.402
Mean absolute error	0.1520	0.1512	0.1537	0.1543
Root mean squared error	0.341	0.324	0.298	0.344
Relative absolute error	70.12%	69.88%	81.23%	74.56%
Root relative squared error	92.12%	90.23%	97.45%	94.76%
TR Rate	0.872	0.905	0.886	0.882
FR Rate	0.514	0.432	0.523	0.549
Precision	0.826	0.893	0.827	0.834
Recall	0.811	0.889	0.854	0.835
F-measure	0.855	0.902	0.824	0.856

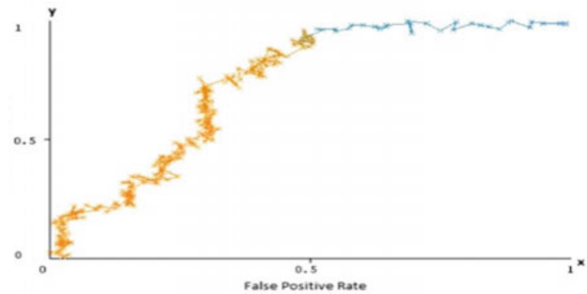


(a)

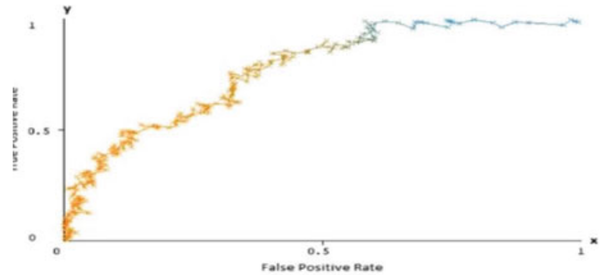
Fig. 2 **a** Logistic regression curve for true positive versus false positive rate. **b** Random forest curve for true positive versus false positive rate **c** Bayes net curve for true positive versus false positive rate **d** KNN curve for true positive versus false positive rate

5 Conclusion

A study on the application of machine learning techniques in the prediction of lung ailments is described in this publication. Random forest outperforms LR, Bayes Net, and KNN. Using data from future classic symptoms, the model might be trained for additional diseases with comparable symptoms. Preprocessing and discretization techniques can be applied to improve the accuracy of the model even further.



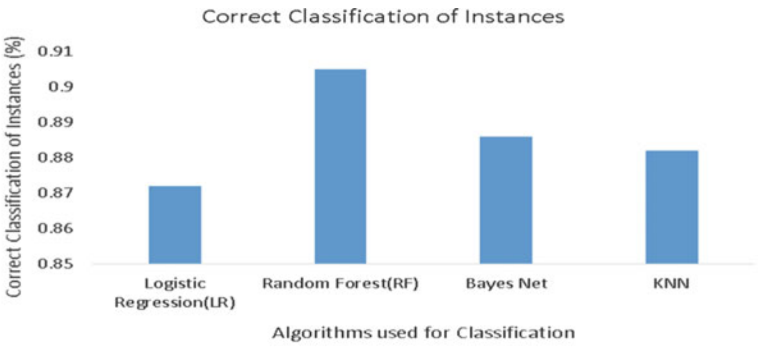
(b)



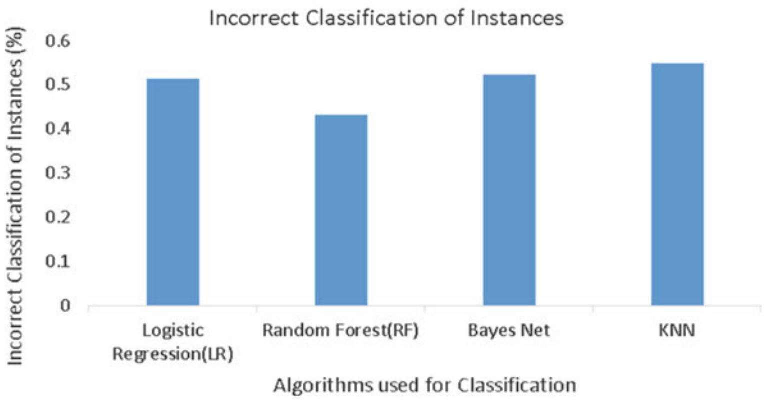
(c)

(d)

Fig. 2 (continued)



(a)



(b)

Fig. 3 a Correct classification of instances done by algorithms. **b** Incorrect classification of instances done by algorithms

References

1. Bharati S, Podder P, Mondal R, Mahmood A, Raihan-Al-Masud M (2020) Comparative performance analysis of different classification algorithm for the purpose of prediction of lung cancer. Advan Intell Syst Comput. Springer. 941: 447–457. https://doi.org/10.1007/978-3-030-16660-1_44
2. Raut S, Patil S, Shelke G (2021) Lung cancer detection using machine learning approach. Int J Advan Sci Res Eng Trends (IJASRET)
3. Moradi P, Jamzad M (2019) Detecting lung cancer lesions in CT images using 3D convolutional neural networks. 4th International conference on pattern recognition and image analysis (IPRIA), pp 114–118
4. Zhu W, Liu C, Fan W, Xie X (12–15 March 2018) DeepLung: deep 3D dual path nets for automated pulmonary nodule detection and classification. In Proceedings of the IEEE winter conference on applications of computer vision (WACV), Lake Tahoe, NV, USA, pp 673–681
5. Senthil S, Ayshwarya B (2018) Lung cancer prediction using feed forward back propagation neural networks with optimal features. Int J Appl Eng Res 13(1):318–325. ISSN 0973–4562
6. Sadiya A, et al (2019) Differential diagnosis of tuberculosis and pneumonia using machine learning. Int J Innov Technol Explor Eng (IJITEE) 8(6S4):245–250. ISSN 2278–3075 In depth analysis of lung disease prediction 213 view publication stats
7. Ismail MBS (2021) Lung cancer detection and classification using machine learning algorithm. Turk J Comput Math Educ 12(13):7048–7054
8. Aykanat M, Kılıç Ö, Kurt B, Saryal S (2017) Classification of lung sounds using convolutional neural networks. EURASIP J Image Video Process 65:1–9. https://doi.org/10.1186/s13640-017-0213-2
9. S. Khobragade, A. Tiwari, C. Y. Pati, and V. Narke (2016) Automatic detection of major lung diseases u sing chest radiographs and classification by feed-forward artificial neural network. In Proceedings of 1st IEEE international conference on power electronics. Intelligent control and energy systems (ICPEICES-2016) 2016 IEEE, Delhi, India, pp 1–5
10. Park DJ, Park MW, Lee H, Kim Y-J, Kim Y, Park YH (2021) Development of machine learning model for diagnostic disease prediction based on laboratory tests. Nat Portfolio Sci Rep 11:7567. https://doi.org/10.1038/s41598-021-87171-5

A New Cascaded H-bridge Multilevel Inverter Using Sinusoidal Pulse Width Modulation

M. Revathi and K. Rama Sudha

Abstract The utilization of multilevel inverter since the last decade has been increased. Due to the ability of these novel inverters to synthesize, the waveforms with better harmonic profile and output are used in number of high voltage and high power applications This paper depicts an asymmetrical 23-level multilevel inverter, using sinusoidal pulse width modulation technique. With increased steps in output voltage, the total harmonic distortion (THD) is reduced. This topology is anticipated to get 23 levels. Simulation results are shown and compared with theoretical results. This topology is proposed to give more number of levels with minimum possible switches, which is more efficient.

1 Introduction

Power is based on non-renewable energy sources. In future, there lies on the use of renewable energy resources to satisfy the power demand increase and needs, and solar energy is most easily available and comfortably used among all renewable energy sources. Available power from solar cell is DC in nature, but for utilization of this, it is indeed to be converted into AC. In India, the demand of electric power is increasing rapidly; at present, the maximum generation of electric power is required for domestic purpose. Inverter is the power electronic modulator that is used for this purpose.

Multilevel cascade-type inverter intakes multiple DC sources and gives combined AC output for desired voltage and frequency. The multilevel inverter has drawn great interest in power industry. With increased number of voltage level better is the voltage waveforms and reduced THD. Multilevel inverters with small output voltage steps result in high voltage capability, less harmonic components, low switching losses, and high power quality. Mostly, a 2-level inverter is used to generate AC. There are

M. Revathi (✉) · K. Rama Sudha
Department of Electrical Engineering, Andhra University College of Engineering(A), Visakhapatnam, Andhra Pradesh, India
e-mail: revathicf@andhrauniversity.edu.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_22

different techniques for reduction the THD and to get sinusoidal by accessible for control and structure of the inverter [1]. A review of reduced power switch count multilevel inverters are presented in references [2, 3]. A new boost multilevel inverter (MLI) topology based on the switched capacitor is proposed in [4]. With reduced switches for a staircase output voltage generations, two new topologies were proposed in reference [5]. A new single-phase MLI topology to obtain higher voltage level at the output and to reduce the number of switches in the circuit is proposed in [6].

- A new configuration of MLI is developed to get 23 levels using only fourteen switches and three DC sources with low THD at the output.
- The proposed 23-level multilevel inverter employs a single carrier sinusoidal pulse width modulation (SCSPWM) scheme to generate the gating signals with triangular wave as carrier.

2 Proposed Topology

The proposed topology 23-level multilevel inverter consists of fourteen switches and three DC voltage sources.

Considering the input voltages and evaluating each level with switches ON for a particular level:

1. First level: Switches S1, S3, S6, S8, S9, S11, and S12 are ON. Thus, the load voltage $V_L = V_{dc}$
2. Second level (2L): Switches S1, S3, S5, S8, S9, S11, and S12 are ON. Thus, the load voltage $V_L = 2V_{dc}$.
3. Third level (3L): Switches S2, S4, S6, S8, S9, S11, and S12 are ON. Thus, the load voltage $V_L = 3V_{dc}$.
4. Fourth level (4L): Switches S1, S4, S6, S8, S9, S11, and S12 are ON. Thus, the load voltage $V_L = 4V_{dc}$.
5. Fifth level (5L): Switches S1, S3, S5, S7, S9, S11, and S12 are ON. Thus, the load voltage $V_L = 5V_{dc.}$
6. Sixth level (6L): Switches S1, S3, S6, S8, S10, S14, and S13 are ON; therefore, voltage across the load becomes $V_L = 6V_{dc.}$
7. Seventh level (7L): Switches S2, S4, S5, S8, S10, S14, and S13 are ON; therefore, voltage across the load becomes $V_L = 7V_{dc}$.
8. Eighth level (8L): Switches S1, S3, S6, S7, S9, S11, and S12 are ON; therefore, voltage across the load becomes $V_L = 8V_{dc}$.
9. Ninth level (9L): Switches S1, S3, S5, S7, S10, S11, and S12 are ON. Thus, the load voltage $V_L = 9V_{dc.}$
10. Tenth level (10L): Switches S2, S4, S6, S9, S11, and S12 are ON; therefore, voltage across the load becomes $V_L = 10V_{dc}$.
11. Eleventh level (11L): Switches S1, S4, S6, S7, S9, S11, and S12 are ON. Thus, the load voltage $V_L = 11V_{dc.}$

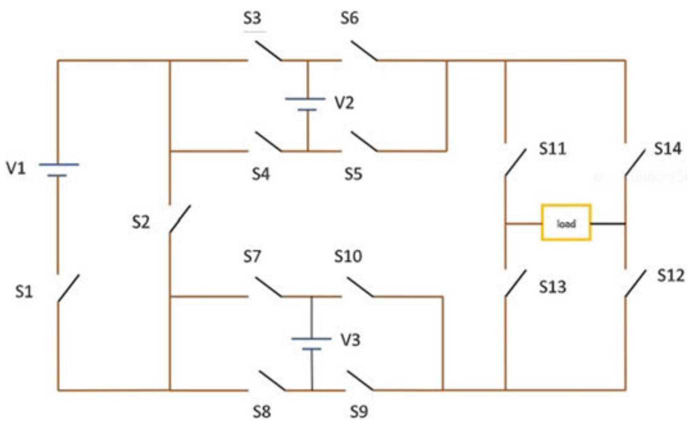


Fig. 1 Topology of 23-level multilevel inverter

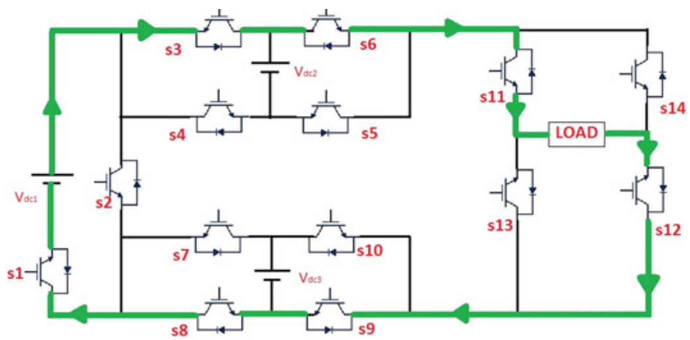


Fig. 2 Level 1($V_o = +1$ V)

Likely, the negative levels of the load voltage waveform are also generated (Figs. 1, 2, 3, and 4; Table 1).

The switching pattern is shown in the table.

3 Simulation Results

On simulating the proposed configuration in MATLAB/SIMULINK, the simulation outputs obtained result in 23 levels and each level with a magnitude of 21 V (Figs. 5 and 6).

It resulted in the total harmonic distortion percentage of 3.69. Hence, the distortion is reduced giving out a better output (Fig. 7).

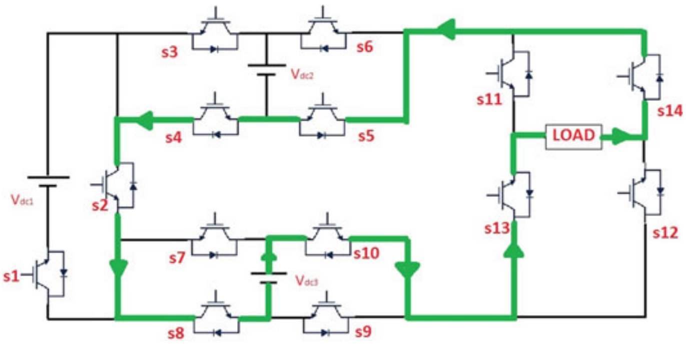


Fig. 3 Level 7($V_o = +7$ V)

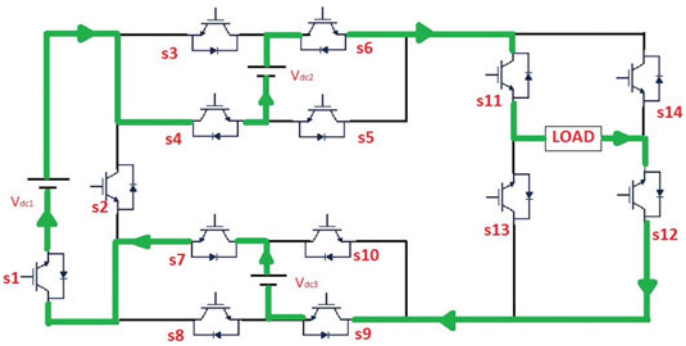


Fig. 4 Level 11:($V_o = +11$ V)

4 Mathematical Calculations

The mathematical relation for the 23-level multilevel inverter can be given as follows: Generalizing the formulae,

Let 'n' be the number of sources and 'm' be an integer.

$$\text{Number of inverter levels (L)} = 2^{(m+n)} + 2^{(mn)} - m$$
$$\text{when } n \text{ is odd, } m = 0, 1, 2\ldots$$
$$= 2^{(n+1)} - 1 \quad \text{when } n \text{ is even}$$

Number of switches:

$$= 2\left[2^{(n-m)} - 1\right] \quad \text{when } n \text{ is odd, } m = -1, 0, 1\ldots$$
$$= \left[2^{n} - m\right]/2^{m} \quad \text{when } n \text{ is even, } m = -1, 0, 1\ldots$$

Table 1 Switching pattern

S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	V0
1	0	0	1	0	1	1	0	1	0	1	1	0	0	11 V
0	1	0	1	0	1	1	0	1	0	1	1	0	0	10 V
1	0	1	0	1	0	0	1	0	1	0	0	1	1	9 V
1	0	1	0	0	1	1	0	1	0	1	1	0	0	8 V
0	1	0	1	1	0	0	1	0	1	0	0	1	1	7 V
1	0	1	0	0	1	0	1	0	1	0	0	1	1	6 V
1	0	1	0	1	0	1	0	1	0	1	1	0	0	5 V
1	0	0	1	0	1	0	1	1	0	1	1	0	0	4 V
0	1	0	1	0	1	0	1	1	0	1	1	0	0	3 V
1	0	1	0	1	0	0	1	1	0	0	0	1	1	2 V
1	0	1	0	0	1	0	1	1	0	1	1	0	0	1 V
0	1	0	1	1	0	1	0	0	1	0	1	0	0	0 V
1	0	1	0	0	1	0	1	1	0	0	0	0	1	−1 V
1	0	1	0	1	0	0	1	1	0	1	1	0	0	−2 V
0	1	0	1	0	1	1	0	0	1	0	0	1	1	−3 V
1	0	0	1	0	1	1	0	0	1	0	0	1	1	−4 V
1	0	1	0	1	0	1	0	1	0	0	0	1	1	−5 V
1	0	1	0	0	1	0	1	0	1	1	1	0	0	−6 V
0	1	0	1	1	0	1	0	1	0	0	0	1	1	−7 V
1	0	1	0	0	1	1	0	1	0	0	0	1	1	−8 V
1	0	1	0	1	0	0	1	0	1	1	1	0	0	−9 V
0	1	0	1	0	1	1	0	1	0	0	0	1	1	−10 V
1	0	0	1	0	1	1	0	1	0	0	0	1	1	−11 V

V1 = 3 V, V2 = 1 V, V3 = 7 V

The topologies referred below are compared with the proposed topology in terms of voltage sources, power switches, diodes, and passive components like capacitors, control circuit, gate driven circuits, etc.

- Cascaded power circuit cell (CPCC)-based MLI [7] uses 4 sources and 10 switches in which 2 are bidirectional switches for 27-level multilevel inverter.
- In topology, highly efficient and reliable configuration-based cascaded MLI (HERC-CMLI) [8] uses 4 sources and 14 switches for 25-level multilevel inverter.
- In topology, Reduced switch-based CHB-MLI [9] uses 4 DC sources and 10 switches in which 4 are bidirectional switches for 17-level multilevel inverter.
- In topology [10], developed cascaded cell (DCC)-based MLI uses 4 input DC sources and 11 switches for 11-level multilevel inverter. Out of 11 switches, 4 switches are for polarity generation and remaining are for level generation (Table 2).

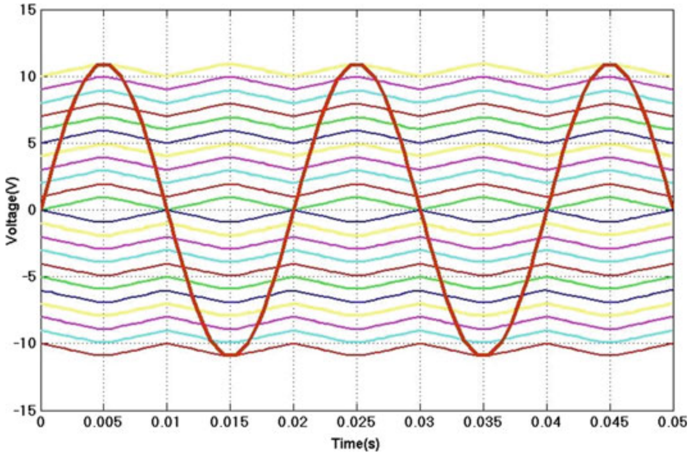


Fig. 5 Triangular carrier waveform with frequency 100 Hz

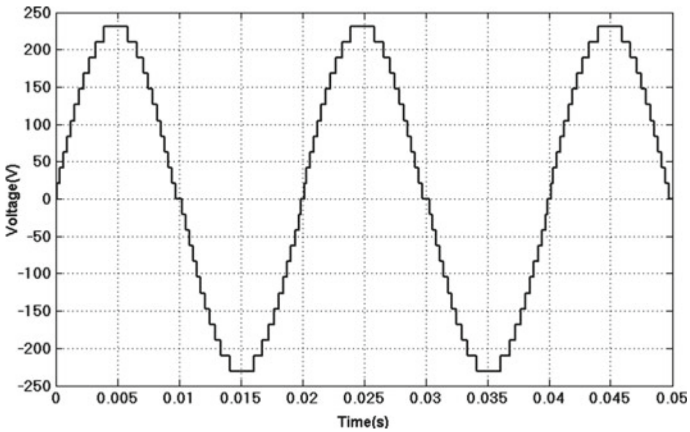


Fig. 6 Simulation output of 23-level MLI

5 Conclusion

With minimum number of switches, the anticipated inverter output voltage inherits the reduction of switching loss and good harmonic performance along with an efficient output. By using proper switching and carrier waveform, the proposed inverter achieved a harmonic profile of 3.69%. An optimal inverter circuit can be developed for reduced switch multilevel inverter topology. With adding an extra H-bridge to the proposed topology each time, there is an increase in levels.

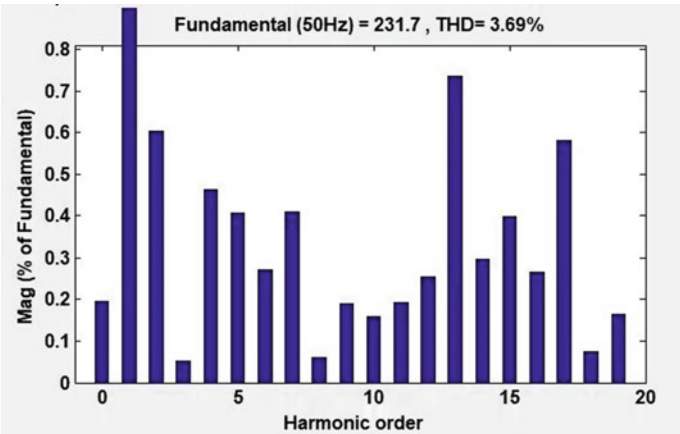


Fig. 7 FFT analysis and THD display of 23-level MLI

Table 2 Comparison with other topologies

MLI's	No. of Levels	No. of voltage sources	No. of switches	No. of diodes	No. of capacitors
1	27	4	12	–	–
2	25	4	14	–	–
3	17	4	12	–	–
4	11	4	11	–	–
Proposed MLI	23	3	14	–	–

References

1. Kamaraja AS, Priyadharshini K (2019) Solar-powered multilevel inverter with a reduced number of switches. Int J Sci Technol Res 8(10):1545–1550
2. Omer P, Kumar J, Surjan BS (2020) A review on reduced switch count multilevel inverter topologies. IEEE Access 8:22281–22302
3. Gupta KK, Ranjan A, Bhatnagar P, Sahu LK, Jain S (2016) Multilevel inverter topologies with reduced device count: a review. IEEE Trans Power Electron 31(1):135–151
4. Siddique MD, Mekhilef S, Shah NM, Ali JSM, Meraj M, Iqbal A, Al-Hitmi MA (2019) A new single phase single switched-capacitor based nine-level boost inverter topology with reduced switch count and voltage stress. IEEE Access 7:174178–174188
5. Siddique MD, Mekhilef S, Shah NM, Memon MA (2019) Optimal design of a new cascaded multilevel inverter topology with reduced switch count. IEEE Access 7:24498–24510
6. Siddique MD, Mekhilef S, Shah NM, Sarwar A, Iqbal A, Tayyab M, Ansari MK (2019) Low switching frequency based asymmetrical multilevel inverter topology with reduced switch count. IEEE Access 7:86374–86383
7. Sadigh AK, Abarzadeh M, Corzine KA, Dargahi V (Dec 2015) A new breed of optimized symmetrical and asymmetrical cascaded multilevel power converters. IEEE J Emerg Sel Topics Power Electron 3(4):1160–1170
8. Jain S, Sonti V (2017) A highly efficient and reliable inverter configuration based cascaded multilevel inverter for PV systems. IEEE Trans Ind Electron 64(4):2865–2875

9. Jayabalan M, Jeevarathinam B, Sandirasegarane T (2017) Reduced switch count pulse width modulated multilevel inverter. IET Power Electron 10(1):10–17
10. Babaei E, Laali S, Bayat Z (2015) A single-phase cascaded multi-level inverter based on a new basic unit with reduced number of power switches. IEEE Trans Ind Electron 62(2):922–929

IoT and Blockchain-Based Method for Device Identity Verification

Chetna Laroiya, Manjot K. Bhatia, Suman Madan, and C. Komalavalli

Abstract The Internet of Things (IoT) is nowadays having outstanding developments in the IT industry and for research purposes. However, it actually experiences security and protection liabilities. Traditional security and protection approaches will, in general, be irrelevant for IoT, principally because of its decentralized geography and the asset imperatives of most of its gadgets. Blockchain that supports the cryptographic money Bitcoin has been as of late used to give security and protection in distributed networks with comparative geographies to IoT. In the IoT, ensuring the validity of the identity of a device accessing the network is the basis of security. A computer network consists of nodes. These nodes are linked through the communication links. The nodes are distributed geographically, and the purpose of the network is to transmit between different types of nodes. These nodes can be personal computers, workstations, sensors, etc. Users carrying wearable devices move from one place to another, and there is a possibility of linking to more than one network. That might raise the anomalies of the information in the network. We aim to remove the anomalies by validating the node with the help of blockchain technology. This paper proposes a system and method for verifying identity using IoT and blockchain technologies. The proposed model will detect anomalies by comparing node traffic profile and behavioral information of the node in the blockchain to the observed node activity.

Keywords Blockchain · Privacy · Confidentiality · Fog computing · Edge computing

C. Laroiya · M. K. Bhatia · S. Madan (✉) · C. Komalavalli
Jagan Institute of Management Studies, Sector 5, Rohini, Delhi, India
e-mail: madan.suman@gmail.com

C. Laroiya
e-mail: chetnalaroiya@gmail.com

M. K. Bhatia
e-mail: manjot.bhatia@jimsindia.org

C. Komalavalli
e-mail: komalalvalli@gmail.com

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_23

1 Introduction

The development of the Internet of Things (IoT) has strong influences on human life. Human lifestyle is steadily changing toward smartness and intelligence which can be enabled by the development of smart homes and smart cities. The Internet of Things (IoT) addresses quite possibly the main problematic technology of this century. It is a characteristic advancement of the Internet (of PCs) to installed and cyber physical frameworks, "things" that, while not clearly PCs themselves, by and by have PCs inside them. With an organization of modest sensors and interconnected things, data assortment on our reality and climate can be accomplished at a lot higher granularity. Without a doubt, such point-by-point information will further develop efficiencies and convey progressed administrations in a wide scope of utilization areas including inescapable medical care and keen city administrations. Nonetheless, the inexorably undetectable, thick, and unavoidable assortment, preparing and dispersal of information amidst individuals' private lives brings about genuine security and protection concerns [1]. From one perspective, this information can be utilized to offer a scope of refined what is more customized administrations that give utility to the clients. Then again, installed in this information will be data that can be utilized to algorithmically build a virtual account of our exercises, uncovering private conduct and way of life designs.

The rapid development of IoT usage and administrations leads to the bottleneck in the communication and correspondence of IoT devices due to the absence of unique identity of the device in the network [2, 3]. The idea of identity of things deals with the allocation of unique identifier to each IoT device and objects ranging from motion sensor to screen conduct trackers, enabling the communication of these devices over the Internet, tracking the relationship as well as the life cycle of objects. Life cycle of the objects might span the person's life or a very limited time span. The electronic medical record (EMR) spans one's life, whereas a parcel being shipped over the Internet has very brief life cycles, till it is shipped. Identity of things also includes addressability and validation strategies. The identity of things is a biggest constituent of IoT. Numerous guidelines are proposed to handle the concerns encountered in device identity management [4, 5]. The customary arrangements for the most part embrace a brought together design, which might prompt a weak link. Then again, versatility is another issue as to upkeep and framework of personality in blockchain environment.

Blockchain innovation started to be known with the coming of crypto coins mining since it is its primary innovation. In any case, presently, it likewise has a lot to do with the IoT. The measure of information prepared by IoT gadgets is gigantic and presented to assaults by cybercriminals. Blockchain engineering is used to verify, normalize, and secure the reception of information taken care of by the gadgets. For IoT security, the blockchain can screen the data gathered by the sensors, without permitting them to be copied by any off-base information. Sensors can likewise move information utilizing blockchain innovation, without the requirement for a confided in outsider [6].

A consortium blockchain is a permissioned blockchain that is made out of multiple organizations instead of a single organization. Multiple organizations share the responsibilities of the blockchain and are well suited for business, wherein all participants need permission. Consortium blockchain, also called as federated blockchain, is much versatile. It is not a public blockchain neither governed by single organization rather it is a permissioned platform which is governed by a group of many organizations. Unlike the hybrid blockchain, it has no public part. Being permissioned blockchain, it guarantees exchange security, as the members validate to the framework prior to initiating the transaction. It confirms less energy utilization when compared with public blockchains by using low-energy consuming consensus protocols for mining process. In this way, consortium blockchain aligns with the need of IoT gadgets and sensors as they too need arrangements with lower-energy utilization.

The integration of blockchain innovation with identity management can be a model solution for many issues which are of public interest, like concentrated control of identities which will lead to the security and accessibility issues, like single point of access and single point of failures of the IoT network. Also, blockchain will wipe out the requirement for confided in outsiders to perform minds' ways of life as the blockchain record is distributed across the consortium members. As the information in BC is permanent, encoded, and cannot be erased, customers will have complete control over their own data. Numerous industry chiefs are quick to take on blockchain to upgrade the protection and security of their character, the executive's foundation, and to change their customary plan of action or to fabricate the economy of sharing idea. In spite of this, the blockchain is not satisfactory for all organizations, and any unstructured or lacking prepared to utilize arrangement might prompt disappointment because of moving the business system for bringing together to the decentralized idea.

The paper proposes a blockchain-based IoT identity management model. Firstly, the proposed architecture describing blockchain-based identity management is shown along with the associated protocol of registration, verification, and revocation of the devices. The proposed protocol addresses the data privacy and security issues in traditional centralized systems. Focus is given on the concept of distributing the authority of an identity system to a group of organizations. Furthermore, we implement the proof-of-concept prototype and evaluate the approach of splitting the main functionalities of an identity management system to separated immutable ledgers and conferred the results. At the end, we give a detailed related work comparison and present our future research directions.

2 Preliminaries

2.1 Related Work

Device authentication is very important in the Internet of Things (IOT), where devices need to communicate and coordinate with each other to perform certain tasks. Secure communication between devices is possible only when identity of the communicating devices is properly confirmed. Considering the features of IoT environment, authentication protocols used in the Internet are not appropriate in IoT. Various device authentication methods exist for IoT, public key infrastructure (PKI) authentication system [7, 8] uses digital certificate for user and device authentication and cryptography for secure communication of devices, certificate-based approach [9] uses large public keys, and complex protocols for issuing more security and certificate less [10]-based schemes are also used by authors for IoT device authentication. The mentioned authentication schemes are centralized and have some drawbacks like security, complexity, and time consuming. Blockchain technology provides decentralized environment and can be used to enhance the security in IoT. Blockchain can be used to register and provide unique identification number to each IoT node. The unique ID assigned by the blockchain is used to uniquely identify each device node and also supports the device to associate with any other device.

In [11], an architecture for the security of IoT is given wherein the author suggested three-tier architecture and included blockchain in between authentication and application layer. Application layer passes the transaction information to the blockchain layer, and blockchain provides device authentication and storage reliability with no increase in cost and service delay. Blockchain is decentralized, but it accepts the centralized device/user authentication method.

In [12], a distributed and decentralized access management system is introduced that manages scalability and access issues of constrained IoT devices. Author proposed decentralized solution, splits IoT devices and blockchain network that makes accumulation of IoT devices easier in the network but not considered the security issues associated with connections. In [13], the new model of IoT architecture is suggested by combining SDN and blockchain technology. Blockchain technique is used to update the flow table securely and verify the version of the table also. After the validation of the flow table, latest table can be downloaded for the forwarding devices. Without human intervention, security adaptation to the threat is achieved in this model. Compared to existing models, this model outperforms with respect to various metrics.

In [14], the fog computing-based authentication scheme is introduced to achieve security against attacks of eavesdropping, replay, and DoS. Ethereum contracts are implemented for testing the overall operation and functionalities of the system. Fog devices are used to balance the work load of IoT devices for authenticating and communicating with the blockchain. In [15], a decentralized device node authentication method is introduced and suggested cloud blockchain for the security of the process. The method satisfies the requirements with respect to security, but authentication delay needs to be considered because of cloud blockchain processing.

2.2 Edge Computing

Edge computing is an umbrella of technologies to take advantage of computing resources available outside the cloud [16]. It is a distributive computing paradigm. Using this technology, workload is placed closer to the origin of data such that appropriate action can be taken in response to the analysis of this data. Edge computing has following intrinsic elements, as shown in Fig. 1:

(a) Cloud: This can be a private or public cloud. It acts as a repository for applications that are used by the different edge nodes [17, 18].
(b) Edge node: It refers to any edge device, edge server, or edge gateway on which edge computing can be performed.
(c) Edge devices: It has limited computing resources, e.g., intelligent camera, ATM.
(d) Edge cluster/server: An edge cluster/server is typically required to run enterprise applications and shared services.
(e) Edge gateway: The edge gateway is an edge cluster/server which has an additional function to perform protocol translation, network termination, and firewall protection.

The edge computing offers several benefits like substantial reduction of latencies at the edge devices, lowering demands on network bandwidth, increasing privacy of

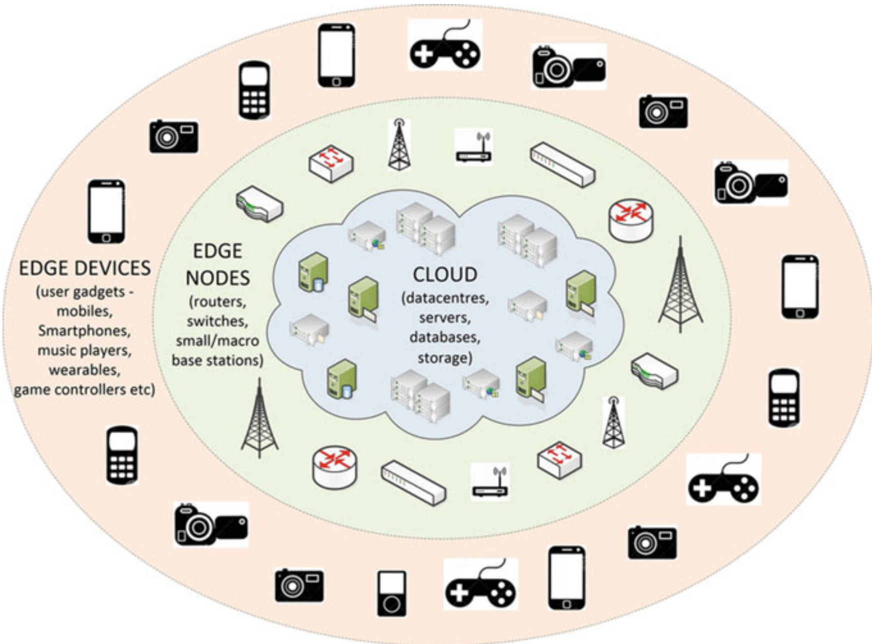


Fig. 1 Edge computing elements

sensitive information, and enabling operations even when networks are disrupted [18, 19]. However, edge computing presents a concern for security at each edge device of the edge network. Not every edge device may have the same built-in authentication and security competences, which is why data becomes more susceptible to breaches. Also, edge computing saves space and cost of communication as useless data is thrown after processing at edge devices and never communicated or saved on cloud. But how can we ensure the right interpretation about the usefulness of data at the edge devices?

Along with the above virtues offered by the technology, it presents a concern for security at each edge device of the edge network. Not every edge device may have the same built-in authentication and security competences, which is why data becomes more susceptible to breaches. Edge computing saves space and cost of communication as pointless data is thrown after processing at edge devices and never communicated or saved on cloud, but the major challenge now lies in the right interpretation about the usefulness of data at the edge devices.

2.3 Blockchain Architecture

Blockchain innovation is one of the exceptionally explored themes in the next decade, because of its appropriated and changeless information stockpiling system empowering applications in practically any region including banking (using consortium blockchain), store network, and other exchange systems like IoT. The idea of blockchain was first presented [20] as the principal innovation of the computerized cryptocurrency called Bitcoin. The utilization of blockchain for Bitcoin exchanges made it the primary cryptocurrency. It was not considered as an exclusive method for advanced cash in a safely and reasonable way. Still there is a flaw called "two-fold spending" in cash transfer. Ordinarily, blockchain innovation is innately impervious to information alteration because of its public record and the agreement system called PoW. Once recorded, information in some random square cannot be adjusted retroactively as this would negate all hashes in the past blocks in a blockchain and break the agreement concurred among hubs voiding the blockchain.

Blockchain can be categorized into three types including public, private, and consortium blockchain [20–22].

Public Blockchain

The public blockchain is a blockchain where anybody can submit exchanges. Anyone can take part in the mining system. In any case, everything hubs can take an interest… in the agreement cycle, and there is an exchange confirmation holding up time, which brings about low exchange throughput. More regrettable, its exchange data is straightforward to the public, which is not helpful for securing protection.

Private Blockchain

The private blockchains have an authorization scheme. Private blockchain identifies which user is writing the platform. This permits the exchange data private. It very well may be viewed as a brought together organization with the previously mentioned weaknesses.

Consortium Blockchain

The consortium blockchain is somewhat decentralized and is mutually settled by different associations. Just some approved hubs partake in the agreement, which works on the efficiency of the agreement and the whole exchange. The charging hub adds the information that should be verified to the related blockchain and stores it for all time to help inquiry whenever. The conditions for choosing blockchain hubs are more extravagant equipment assets or better working climate, which can further develop the execution efficiency of the framework and guarantee the security of the framework.

Blockchain can be fundamentally considered as a chain formed information structure in which a chain of squares is associated with one another through a location pointer dependent on a hash esteem; i.e., blockchain is a common, decentralized, and dispersed state machine. This implies that all hubs freely hold their own duplicate of the blockchain, and the current known “state” is determined by handling every exchange all together as it shows up in the blockchain. Each square of a blockchain regularly contains six sections: hash of the past block, nonce (“number utilized once”), the hash of the current square, Merkle root (hash of various exchanges), timestamp, and exchange information as shown in Fig. 2.

With regard to Bitcoin, exchanges by and large comprise the sender’s location, beneficiaries address, and the worth. Notwithstanding, contingent upon the application this can differ. The header of each square contains a bunch of metadata that assists with approving each square and connection to past blocks in the public record.

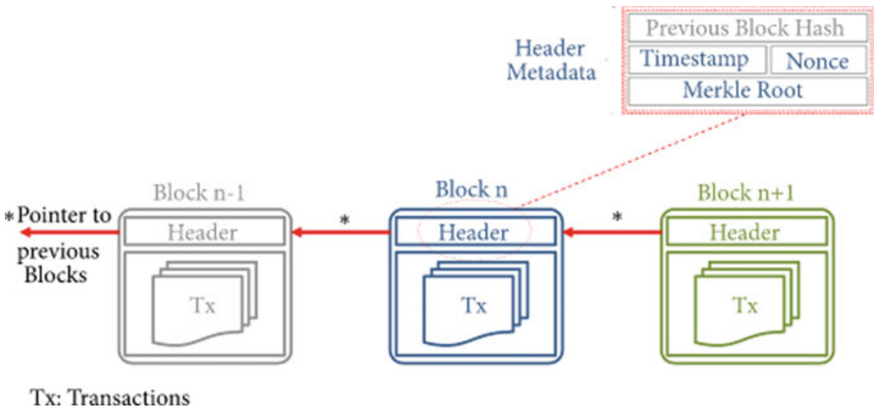


Fig. 2 Blockchain architecture

Having a public record implies that the information and admittance to the framework are accessible to any individual who will take an interest (e.g., Bitcoin, Ethereum, and Litecoin blockchain frameworks).

Notwithstanding, contingent upon the application prerequisites, the construction of the blockchain can be planned in either a more concentrated or a more decentralized way [23, 24]. In such a manner, private blockchain structures are more integrated as they are inhibited by a brought together position that controls the admittance to the blockchain network. Like private blockchains, consortium blockchains are constrained by a bunch of chosen hubs instead of one explicit association, thus a reasonable contender for IoT applications.

3 Proposed Method

In this section, we present our proposed system and model for verifying device's identity using IoT and blockchain technology. The proposed method is used for verifying identity of the devices with the help of public key so that anomalies in the devices can be detected easily. The node information is shared between different blockchain servers for the detection of behavioral changes of the node. It helps to locate and detect the malfunctioning or malicious node in the network. Since blockchain blocks are immutable, thus it ensures the valid information about the particular node at any point of time. Figure 3 shows the block diagram of proposed model.

The proposed model consists of combination of hardware, software, human ware, and human operators. Several fog/node devices are connected to the edge devices. Edge devices are connected to the blockchain server over LAN or WAN and are registered with the blockchain server. These blockchain servers are also connected with each other in a peer–peer manner for exchanging their information. For instance, nodes 1 and 2 may be registered with edge device. Node 1 is forming a first local network, and nodes 3 and 4 may be registered with edge device 2 forming a second local network. New node tries to connect with the network for the first time and sends the registration request to the edge device. Identification and authentication of IoT devices are very important before communication with any other device in the network. Different organizations and groups have provided ways to uniquely identify IoT devices. Some companies have provided IDs like barcodes, IMEI, or RFIDs as unique IDs to the IoT devices, but still it is difficult to say that some common identification scheme exists for IoT identification. Here in this paper, we are using identification (ID) assigned by the manufacturer to the device as the unique ID of IoT node. The nodes IDs as ID1, ID2,…, IDn are used for registration of IoT nodes with edge node. Edge devices are also identified by unique IDs as EID1, EID2,…, EIDn and are already registered with the blockchain server with unique IDs as EIDs.

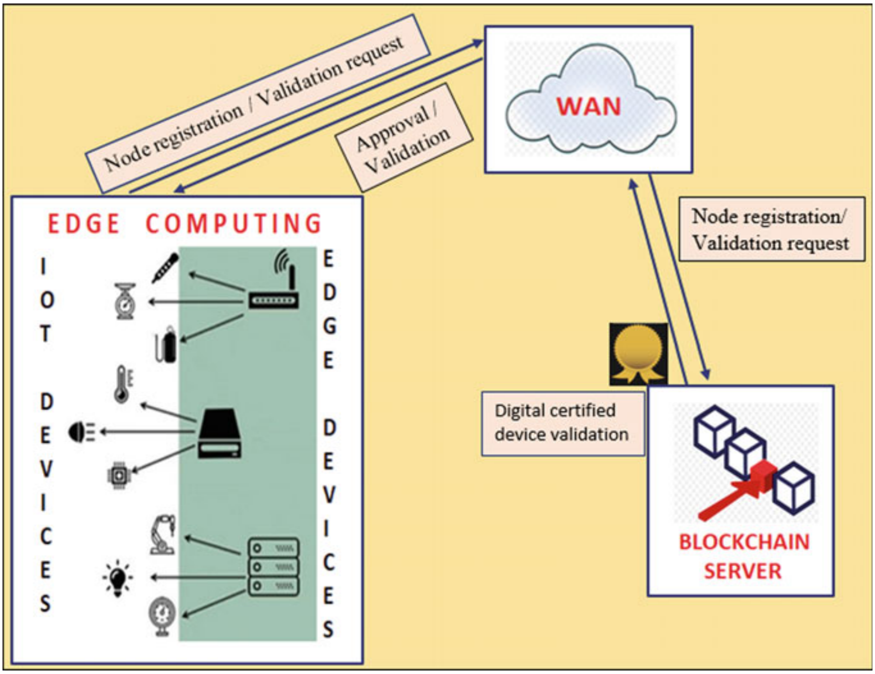


Fig. 3 Block diagram of proposed model

3.1 Phases Involved in Proposed System

The entire model consists of three phases: registration phase for edge device, authentication phase, and validation phase. The process involved in each phase is explained below.

Registration Phase:

1. Each IoT node passes its ID to the edge node for its registration.
2. Edge device will add its own (EID) with ID of the IoT node (ID, EID).
3. Edge device encrypts the (ID, EID) using symmetric encryption mechanism and sends it to blockchain server for verification.
4. Blockchain server verifies the already registered edge node (EID_i) and after verification stores the details of associated IoT node.
5. Once the ID of the IoT node associated with the verified edge node is inserted in the blockchain server as a block, registration is completed.

Authentication Phase:

1. For authentication process, IoT node sends its ID to the edge device.
2. Edge device adds its EID, encrypts (ID, EID), and passes to blockchain server for authentication.

3. Blockchain server compares encrypted (ID, EID) received from edge device with its existing block.
4. If it matches, node is authenticated.
5. Blockchain validates the node with the existing information and any changes in the domain that will be notified to the blockchain server.
6. Blockchain server gets updated with the new domain information of the node and assigns a high-level trust to each node in the chain.

Validation Phase:

1. The validator will then compare the data about node X to the blockchain to ensure that it is valid in light of what is currently known about node X in the chain.
2. Blockchain gets verified and broadcasted to all other servers of the network so as to access the node details by other devices in the network.
3. When a node migrates to the other network (when a node X sends a message to another node, etc.) and identifies anomalies (e.g., by matching traffic profile information or other behavioral information about the node X stored in the blockchain to the node X's observable behavior), and to execute other tasks using the node X's mutual information.
4. Blockchain server is updating regularly with the support of edge devices in the local network. These devices can keep track of the node's behavioral information such as number of bytes sent, size of the data, and destination, etc., and updates the blockchain server accordingly. This helps in deciding about the normal or abnormal behavior of the node
5. With the help of nodes information in the blockchain, devices of IoT network can be monitored for their behavior. Node with a low level of trust can be prevented for communication with certain devices and can be monitored for the authentication also.
6. If any node moves from one network into another, it becomes closer to the second edge device network. As part of this process, edge devices can validate the node with the help of a blockchain server by decoding the digitally signed information using a public key.
7. Stored behavioral information about the node can be used by the device 2 for the detection of anomalies. Comparing the traffic profile of edge device 1 and 2, device 2 can take the appropriate action in the case of anomalies detection.

4 Conclusion

IoT becomes an integral part of our day-to-day life, fueled mainly by the convenience and efficiency it brings. For the security of the generated data and to avoid impersonation attacks, proper node identification is required. In this paper, we have proposed a system and method for verifying the identity of the IoT device using blockchain technologies. The proposed method and model combine the security based on trust model and cryptography. Firstly, our proposed model stores the data about the node by sending their request to the edge device. Secondly, based on the request, the data is stored in the blockchain for further validation. Validated data is stored as a block and broadcasted to all other servers of the network. Our model detects the anomalies of any node moving from one network to another network by validating the data against stored blockchain block. Behavioral changes can also be identified with the help of blockchain block, and appropriate action can be taken by the edge devices. In the future, context-based identity can be incorporated for a better management and will be building prototype that is more robust in nature and based on real-time scenarios.

References

1. Das ML (2015) Privacy and security challenges in internet of things. Distributed computing and internet technology, pp 33–48
2. Hu P, Dhelim S, Ning H, Qiu T (2017) Survey on fog computing: architecture, key technologies, applications and open issues. J Netw Comput Appl 98:27–42
3. Ning H, Ye X, Sada AB, Mao L, Daneshmand M (2019) An attention mechanism inspired selective sensing framework for physical-cyber mapping in internet of things. IEEE Internet Things J 6:9531–9544
4. Ning H, Liu X, Ye X, Zhang JHW, Daneshmand M (2019) Edge computing based ID and nID combined identification and resolution scheme in IoT. IEEE Internet Things J 6:6811–6821
5. Hu P, Ning H, Qiu T, Xu Y, Luo X, Sangaiah AK (2018) A unified face identification and resolution scheme using cloud computing in internet of things. Future Gener Comput Syst 81:582–592
6. Del-Valle-Soto C, Durán-Aguilar G, Cortes-Chavez F, Rossa-Sierra A (24–28 July 2019) Energy-efficient analysis in wireless sensor networks applied to routing techniques for internet of things. In: Proceedings of the international conference on applied human factors and ergonomics. Washington, DC, USA, pp 312–321
7. Alfandi O, Bochem A, Kellner A, Hogrefe D (2013) Improving energy efficiency of data communication in a hybrid PKI-based approach for WSNs. In: Processing of 2013 IEEE 10th consumer communications and networking conference (CCNC)
8. Kadri B, Feham M, Hamed AM (2010) Lightweight PKI for WSN μPKI. J Sec Commun Networks 10(2):135–141
9. Prasad M, Manoharan R (2017) A secure certificate based authentication to reduce overhead for heterogeneous wireless network. In: Proceedings of international conference on advanced computing and communication systems
10. Nong Q (2018) Practical secure certificateless cryptographic protocol with batch verification for intelligent robot authentication. In: Proceedings of international conference on mechatronics and intelligent robotics, pp 483–488
11. Bao Z, Shi W, He D, Chood KR (2018) Iotchain: a three-tier blockchain-based iot security architecture. arXiv preprint arXiv:1806.02008
12. Novo O (2018) Blockchain meets IoT: an architecture for scalable access management in IoT. IEEE Internet Things J 5(2):1184–1195
13. Sharma PK, Singh S, Jeong Y, Park JH (2017) Distblocknet: a distributed blockchains-based secure SDN architecture for IoT networks. IEEE Commun Mag 55(9):78–85
14. Almadhoun R, Kadadha M, Alhemeiri M, Alshehhi M, Salah K (2018) A user authentication scheme of IoT devices using blockchain-enabled fog nodes. In: Proceedings of 2018 IEEE/ACS 15th international conference on computer systems and applications (AICCSA)

15. Hammi MT, Hammi B, Bellot P, Serhrouchni A (2018) Bubbles of trust: a decentralized blockchain-based authentication system for IoT. Comput Secur 78:126–142
16. Cao K, Liu Y, Meng G, Sun Q (2020) An overview on edge computing research. IEEE Access 8:85714–85728. https://doi.org/10.1109/ACCESS.2020.2991734
17. Madan S, Goswami P (2019) k-DDD measure and mapreduce based anonymity model for secured privacy-preserving big data publishing. Internat J Uncertain Fuzziness Knowl-Based Syst 27(02):177–199. https://doi.org/10.1142/S0218488519500089
18. Madan S, Bhardwaj K, Gupta S (Aug 2021) Critical analysis of big data privacy preservation techniques and challenges. In: International conference on innovative computing and communications: proceedings of ICICC 2021, volume 3, vol 1394. Springer Nature, p 267
19. Madan S, Goswami P (2021) Hybrid privacy preservation model for big data publishing on cloud. Int J Advan Intell Paradigms 20(3–4):343–355. https://doi.org/10.1504/IJAIP.2021.119022
20. Nakamoto S (2008) Bitcoin: a peer-to-peer electronic cash system
21. Casino F, Dasaklis TK, Patsakis C (2019) A systematic literature review of blockchain-based applications: current status, classification and open issues. Telematics Inform 36:55–81. https://doi.org/10.1016/j.tele.2018.11.006
22. Zheng Z, Xie S, Dai H-N, Wang H (2016) Blockchain challenges and opportunities: a survey Work Pap
23. Komalavalli C, Saxena D, Laroiya C (2020) Overview of blockchain technology concepts, handbook of research on blockchain technology. Academic Press, pp 349–371, ISBN: 9780128198162
24. Laroiya C, Saxena D, Komalavalli C (2020) Applications of blockchain technology, Academic press, pp 213–243, ISBN: 9780128198162

Designing Intelligent Intrusion Detection System for Industry 4.0 Using Feature Learning Techniques

Sunil Kaushik and Akashdeep Bhardwaj

Abstract Increase in connectivity and cost pressure has pushed Industry 4.0 to rely on the systems built over Internet of Things (IoT). These IoT devices are susceptible to cyber-attacks. Intrusion detection system (IDS) protects such IoT devices from such attacks. However, the IoT devices are wreaked with the high-computational costs and curse of dimensionality. Current study presents an intelligent IDS system which is able to reduce the unwanted features. Proposed IDS system shows an accuracy of 99.26% with a precision over 99% to identify the attacks from the CICIDS2018 dataset.

1 Introduction

Latest advancement in the technology has made everyone dependent on the computer networks. Computer networks have not only become the back bone of the digital information but also have become vital mechanism to provide various sources ranging from bill payments to handling an autonomous vehicle [1, 2]. The increasing dependence on computer networks has brought the computer networks on a sweat spot for the hackers and intruders. The Purplesec report of 2019 [3] suggests that there is an increase in the number of cyber-attacks on yearly basis. The cyber-attacks not only cause inconvenience but also cost fortune to an organization. According to CISCO Annual Security Report 2020, it costs approximately $3.92 M to a company for cyber-attacks [4]. Hence, intrusion detection systems (IDS) are proposed for detecting and preventing the cyber-attacks [5, 6].

S. Kaushik (✉) · A. Bhardwaj
University of Petroleum and Energy Studies, Dehradun, India
e-mail: sunil.kaushik2011@gmail.com

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_24

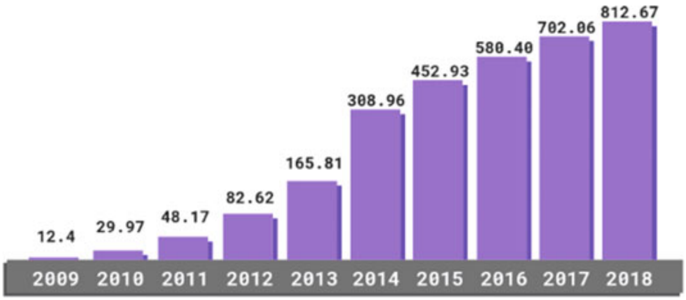


Fig.1 Number of cyber-attacks per million [3]

Figure 1: Number of cyber-attacks per million [2]

This research discusses a new industrial IDS system exploiting the feature selection techniques. The main contributions of this study are as follows:

1. Novel intelligent feature learning mechanism.
2. Novel IDS system using the new feature learning techniques and efficient machine learning (ML) classifier yielding higher accuracy.

2 Review of Literature

Ahuja et al. [6] proposed a NIDS system which is based in the two layers of ensemble feature selection techniques using XGBoost and neural networks (NN) and various machine learning algorithms. They were able to show an accuracy of 96% and AUC of 96.

Bedi et al. [7] used the KDDCup99 data with various ML classifiers such as J48 DT and SVM with the PSO as feature selection algorithm. They were able to achieve an accuracy of 90%.

Kumari and Mehta [8] used the datasets with CICIDS2018, UNSW-NB15, ISCX2012, and NSL-KDD with various ML classifiers and detected accuracy of 99%

Kilincer et al. [9] used SVM as feature-based feature selection technique in tandem with the ML algorithms and were able to achieve accuracy of 96%.

Jan et al. [10] used stacking classifier of XGBOOST, DT, and LR with feature selected using the Chi-Square method on CICIDS2018 dataset. The proposed model produced the accuracy of 98.8% and F1-score of 97.9%.

Fitni et al. [11] used three layers of auto encoders for feature selection and random forest as the classifier. They were able to achieve the accuracy of 99% on their own dataset. Also, they have tried making the whole system light weight and achieved good accuracy in less time using the KNN algorithm.

Latah and Toker [12] presented an IDS system using an ensemble of KNN and extreme learning machine (ELM) for feature selection and DT, RF, NB, and SVM

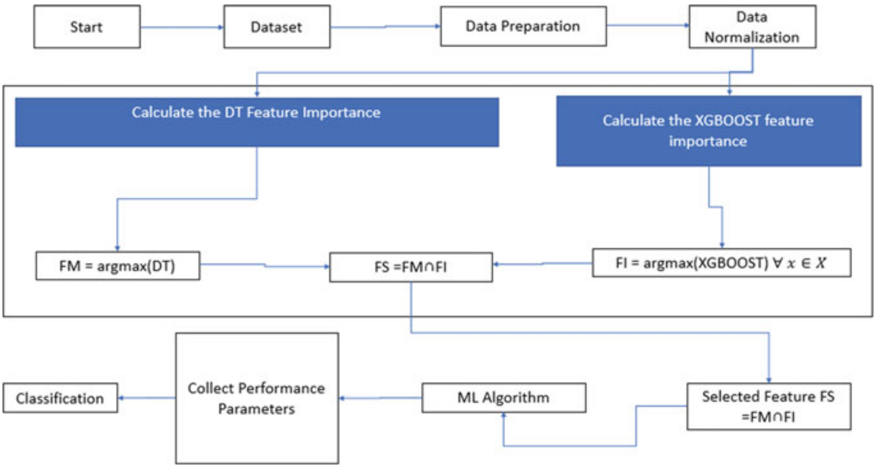


Fig. 2 Experimental setup and proposed algorithm

classifiers on CICIDS2018 dataset. They were able to detect the maximum accuracy of 96.45%.

Sah and Banerjee [13] selected 11, 12, 13, and 15 feature selected using NSL-KDD data with ML algorithms such as NB, KNN, RF, and SVM and were able to achieve maximum accuracy of 99.63% using RF.

3 Research Methodology and Experimental Setup

This section introduces the methodology followed and experiment setup to validate the proposed model and to propose a suitable classifier for the IDS system. In the current research, we applied the data preprocessing steps to remove the not identified values, applied the transformation, and normalization to ensure that good data leads to proper results. Features were selected using the DT and the XGBOOST feature importance and features thus extracted were applied to various machine learning algorithms. The performance of ML algorithms was evaluated based on the criteria given. On a high level the complete experimental setup is depicted in Fig. 2.

4 Results

This section talks about the results obtained during the experiments. The experiments were carried out with the original data with full features and were processed using techniques such as filling the missing values, not identified values, and other processing techniques described earlier. The processed and normalized dataset was applied to a few important classifiers whcih are given in Table 1.

The new feature selection framework was applied to the dataset and important features were selected (Table 2). A newly built dataset by selecting data from CICIDS2018 dataset for features selected through proposed algorithm was created and was applied to the classification algorithms and performance parameters were obtained (Table 3). This section shows that new feature set reduced the training time and testing time of all the algorithms considerably while increasing or retaining the accuracy, precision, recall, and F1-score for all the classification algorithms.

Table 1 suggests that NB and SVM took the least and highest training time with values of 0.05 and 11.3 s. The DT, RF, and LDA took reasonable time of 0.67, 0.58, and 0.58 s, and the LR and LDA took the training time which is on higher side. Also, these ML algorithms showed the same pattern in training time with reduced feature set. Least and maximum time of 0.02 and 8.41 s were shown by NB and SVM, respectively. The maximum reduction in the training time is shown by LR at 63% followed by NB at 60%. RF showed the minimum reduction in training time with 24%. LDA, DT, SVM, and GBM showed the reduction of 50, 46, 24, and 30%.

The proposed feature selection algorithm was applied to the dataset and important features were selected (Table 2). A newly built dataset by selecting data from CICIDS2018 dataset for features selected through proposed algorithm was created and was applied to the classification algorithms and performance parameters were

Table 1 Performance parameters with initial dataset

Classifier name	Train time	Test time	Accuracy	Precision	Recall	F1-score
LR	5.02	0.03	90.6	90.7	90.15	90.43
LDA	0.58	0.018	86.8	86.6	86.8	86.8
NB	0.05	0.02	71.19	75.2	73.29	74.19
DT	0.67	0.013	99.21	99.21	99.21	99.21
RF	0.58	0.029	99.26	99.26	99.26	99.26
SVM	11.32	0.38	76.3	76.13	76.33	76.26
GBM	7.95	0.06	98.12	98.2	98.11	98.12

Table 2 Features selected through proposed algorithm.

Feature selected by proposed algorithm.
’Idle Mean’,’PSH Flag Count’, Bwd Packet Length Mean’, ’Bwd Packet Length Std’, ’FIN Flag Count’, ’Active Min’, ’Init_Win_bytes_backward’, ’Bwd IAT Mean’, Fwd Packet Length Max’, ’Active Mean’, ’Bwd Packet Length Min’, ’Packet Length Mean’, ’Average Packet Size’, ’Fwd IAT Total’, ’Max Packet Length’, ’Flow IAT Min’, ’Min Packet Length’, ’Fwd Packet Length Std’, ’Flow IAT Mean’,’Total Length of Bwd Packets’, ’Bwd Packets/s’, ’Fwd Packet Length Min’, ’Flow Bytes/s’, ’Total Backward Packets’, ’Total Fwd Packets’, ’Down/Up Ratio’, ’Bwd Header Length’, ’Fwd IAT Max’, ’Fwd Packet Length Mean’, ’Init_Win_bytes_forward’, ’Fwd IAT Min’, ’Total Length of Fwd Packets’, ’Bwd IAT Min’, ’Fwd IAT Std’, ’Fwd Header Length’, ’URG Flag Count’, ’ACK Flag Count’, ’Active Max’, ’Flow Duration’, ’Packet Length Std’

Table 3 Performance parameter for the selected feature dataset

Classifier name	Train time	Test time	Accuracy	Precision	Recall	F1-score
LR	1.83	0.01	90.92	90.92	90.92	90.92
LDA	0.29	0.01	88.76	88.76	88.76	88.76
NB	0.02	0.011	75.22	75.22	75.22	75.22
DT	0.36	0.007	99.31	99.31	99.31	99.31
RF	0.44	0.021	99.3	99.3	99.3	99.3
SVM	8.41	0.31	79.19	79.19	79.19	79.19
GBM	5.49	0.05	98.81	98.81	98.81	98.81

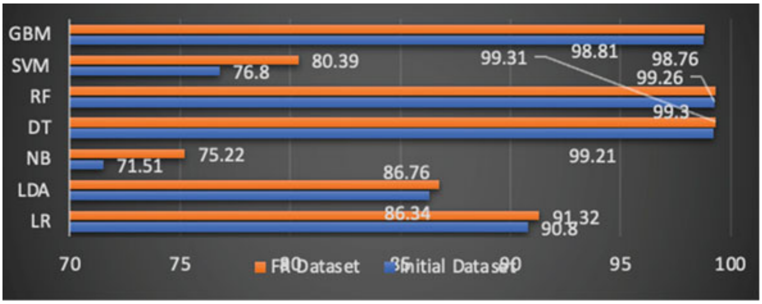


Fig. 3 Accuracy comparison

obtained (Table 3). Figure 3 shows that the proposed algorithm reduced the training time and testing time of all the algorithms considerably while increasing or retaining the accuracy, precision, recall, and F1-score for all the classification algorithms. The proposed algorithm reduced the training time by approximately in the range of 27% to 63%, and a similar trend was seen in the case of testing time. DT showed maximum reduction in both times with the proposed MI2G algorithm, while the SVM showed the least reduction in given parameters.

As shown in Fig. 3, DT and RF classifiers showed the maximum accuracy of 99.3% followed by GBM at 98.8% with feature reduced dataset. All the classifiers showed an increase in the accuracy with FR dataset. The increase in accuracy was in range of 0.04–5.66%. The maximum increase in the accuracy was shown NB classifier, while RF showed the lowest increase in the accuracy. LR and DT showed a marginal improvement in the accuracy with the FR dataset but the classifiers such as LDA, SVM, and GBM showed the significant increase in the accuracy as 2.25, 3.78, and 1.82%.

Similar to accuracy, all the classifiers showed an increased precision (Fig. 4.) when applied to FR dataset. The DT and RF showed the highest precision with initial and FR dataset. DT and RF showed precision of 99.2% with the initial dataset and 99.3% with the FR dataset. GBM showed comparable precision at 98.8% with the FR dataset and 98.12% with the initial dataset. SVM showed highest increase in the precision

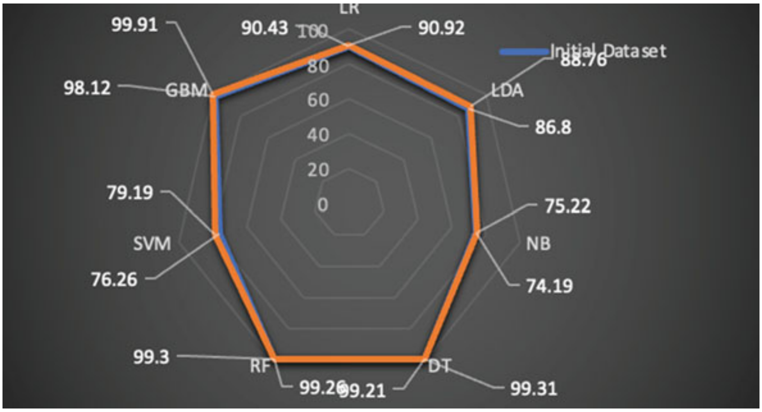


Fig. 4 Comparison of precision score in initial and feature reduced dataset

with the FR dataset and NB showed the least increase in the precision. SVM and NB showed the precision of 79.19% and 75.22% with the FR dataset and 76.13% and 75.2%, respectively for initial dataset. LR and LDA showed the precision of 90.7% and 80.6% with the initial dataset. This was increased by 0.25% and 2.4%, respectively for the FR dataset.

NB and SVM classifier showed the recall of 73.29% and 76.33% with the initial dataset which is among the lowest recall shown by all the classifiers (Fig. 5). However, these two classifiers showed the maximum increase of 2.25% and 3.74%, respectively in the recall with the FR dataset and showed the recall of 75.22% and 79.19%. On the other hand, DT and RF showed the maximum recall around 99.2% and showed the minimum increase in the range of 0.04–0.1% with the FR dataset. GBM showed a recall of 98.11 and 98.81 with initial and FR dataset, respectively which is fairly good in comparison with other classifiers. The rest of the classifiers LR and LDA

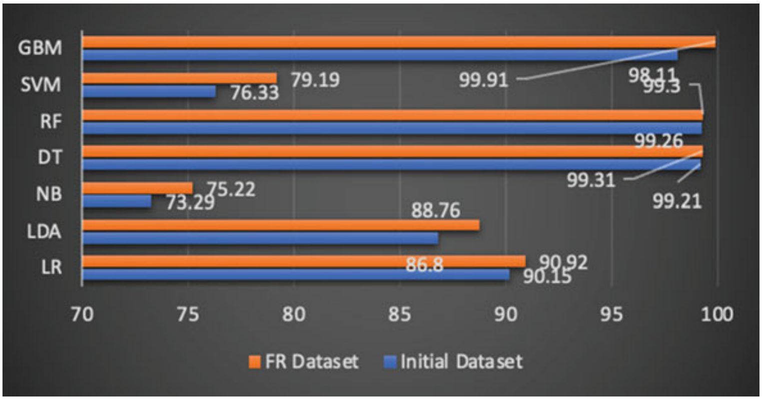


Fig. 5 Comparison of recall in initial and feature reduced dataset

showed recall 90.15% and 86.8%, respectively with the initial dataset and 90.92% and 88.76%, respectively with the FR dataset.

Figures 6 and 7 suggest that DT and RF classifiers with selected features using the proposed algorithm score highest on all the performance parameters. The high accuracy with high precision suggests that classifiers can differentiate between attacks and normal vectors efficiently and is stable enough to predict the labels with very few misses or low false positive rate. The high recall with good accuracy and precision suggests that DT and RF both did not overfit or underfit the data and provided a low false negative rate. In addition, the reduction in training time and testing time of classifiers with selected features suggests that new feature selection algorithms can select the features which can yield important and relevant information and can

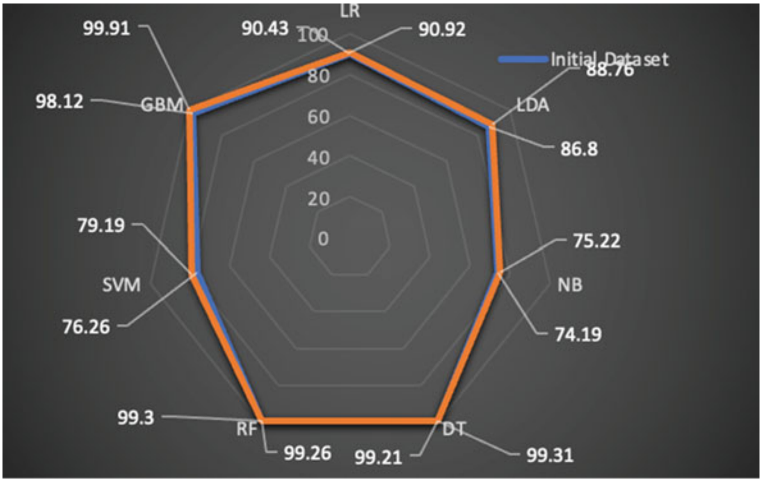


Fig. 6 Comparison of F1-score of initial dataset and feature reduced dataset

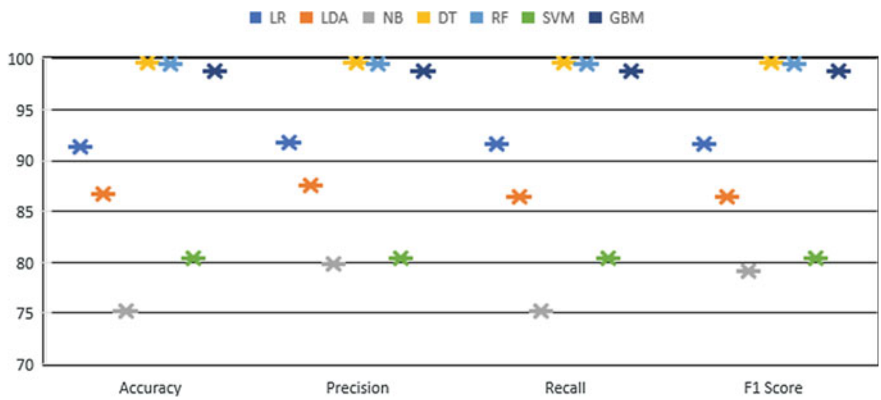


Fig. 7 Comparison of all classifiers on various parameters on reduced features

remove the bias and noise from the data. The selected features can increase the accuracy, precision, recall, and F1-score with a huge reduction in training and testing time with all of the classifiers. DT can show the least training time with high-performance parameters. This leads to the conclusion that proposed algorithm with DT and RF can be considered as a potential choice for an IDS system working on industrial IoT systems.

5 Conclusion and Future Works

This study proposed a new feature selection algorithm and tests the effectiveness of the algorithm with various classifiers. Study finds out the DT and new algorithm provides the accuracy and precision of 99% and are best suitable to be used as IDS for IoT systems. This study can further be augmented with the neural networks to increase the accuracy of attack detection with fairly high precision in lesser time.

References

1. Makuvaza A, Jat DS, Gamundani AM (2021) Deep neural network (DNN) solution for real-time detection of distributed denial of service (DDoS) attacks in software defined networks (SDNs). SN Comput Sci 2(2):1–10
2. Hadem P, Saikia DK, Moulik S (2021) An SDN-based intrusion detection system using SVM with selective logging for IP traceback. Comput Netw 191:108015
3. Purple Sec (2021) Trends. https://purplesec.us/cyber-security-trends-2021/. Accessed on 15 Dec 2021
4. 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
5. Thomas RN, Gupta R (2021) Design and development of an efficient network intrusion detection system using machine learning techniques. Wireless Commun Mob Comput (Online)
6. Ahuja N, Singal G, Mukhopadhyay D, Kumar N (2021) Automated DDOS attack detection in software defined networking. J Netw Comput Appl 187:103108
7. Bedi P, Gupta N, Jindal V (2021) I-SiamIDS: an improved Siam-IDS for handling class imbalance in network-based intrusion detection systems. Appl Intell 51(2):1133–1151
8. Kumari A, Mehta A(2020) A hybrid intrusion detection system based on decision tree and support vector machine. In: 2020 IEEE 5th international conference on computing communication and automation (ICCCA), pp 396–400
9. Kilincer IF, Ertam F, Sengur A (2021) Machine learning methods for cyber security intrusion detection: datasets and comparative study. Comput Netw 188:107840
10. Jan SU, Ahmed S, Shakhov V, Koo I (2019) Toward a lightweight intrusion detection system for the internet of things. IEEE Access 7:42450–42471
11. Fitni QRS, Ramli K (2020) Implementation of ensemble learning and feature selection for performance improvements in anomaly-based intrusion detection systems. In: 2020 IEEE international conference on industry 4.0, artificial intelligence, and communications technology (IAICT), pp 118–124
12. Latah M, Toker L (2020) An efficient flow-based multi-level hybrid intrusion detection system for software-defined networks. CCF Trans Netw 3(3):261–271

13. Sah G, Banerjee S (2020) Feature reduction and classifications techniques for intrusion detection system. In: International conference on communication and signal processing, pp 1543–1547

Optimizing Job Scheduling Problem Using Improved GA + CS Algorithm

Sudhanshu Prakash Tiwari and Gurpreet Singh

Abstract Soft computing-based several techniques had already been applied previously in various industrial applications. This paper tries to provide application of various algorithms on job scheduling problem. The expected future needs of industry are based on proper application of these algorithms. In single objective optimization, point is to discover a schedule that limits general culmination time known as makespan. The paper represents comparative study of algorithms shows calculation of reduced makespan. Modified computation allocates jobs precisely than GA. Thus after applying, it is found that performance of hybrid genetic-cuckoo search algorithm approach is effective in finding ideal solutions contrasted with that of different methodologies.

Keywords Genetic algorithm · Makespan · Cuckoo search

1 Introduction

Job scheduling [1–3] is process of allotting resources to jobs. Issue related to scheduling emerged for a situation where resource number is lesser than jobs and becomes critical issue. Fundamental assignment of scheduling provides jobs required resources in such a way that other jobs will not be hampered. Numerous calculations have been connected to job shop scheduling issue. All metaheuristic calculations [4–6] have been connected to take care of job shop scheduling issue. Genetic algorithm gives better outcomes as contrast other metaheuristic calculations, however, genetic algorithm turns out to be effortlessly bind with nearby optima. To stay away from this issue, we have utilized CS calculation. CS calculation is effective as well as quick calculation. It utilizes a solitary parameter for searching. Hybrid genetic with

S. Prakash Tiwari (✉)
Mewar University, Gangarar, Rajasthan, India
e-mail: Sudhanshu.iiitg@gmail.com

G. Singh
CSE, Punjab Institute of Technology, Rajpura (MRSPTU, Bathinda), India

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_25

CS is utilized to unravel scheduling issue. It tends to be utilized in different modern issues to unwind different combinatorial improvements issues. The noteworthy issue in scheduling is that a significant number of them do not fit into a typical depiction demonstrate. So defining a common framework is difficult for scheduling.

1.1 Genetic Algorithm

Genetic algorithms (GAs) [7] are an improvement philosophy in view of an immediate similarity to Darwinian common determination in natural generation. Prompted varieties and recombination of these ideas are tried against an assessment capacity to see which one will make due to the individual to come.

The utilization of genetic calculations requires the accompanying segments:

- A method of encoding answers for the issue
- An assessment task that profits a rating for every solution
- A method of introducing the number of inhabitants in solutions.

Genetic calculations with straightforward recombination operators have been utilized in job shop scheduling. Their significance on relative requesting composition, outright requesting pattern, cycles, and edges in the offsprings emerge contrasts in such visually impaired recombination operators. Sequencing issues have been tended to by mapping their requirements to a Boolean fulfillment issue utilizing incomplete result designs. This example has delivered great outcomes for extremely straightforward issues. Diverse sorts of heuristic genetic calculations have been connected to job shop scheduling. In these genetic plans, issue particular heuristics are fused in the recombination operators.

Researchers have utilized genetic calculations to understand a double criteria job shop scheduling issue in a genuine creation office. Those criteria were the minimization of normal stock in the plant and the minimization of the normal sitting tight time for a request to be chosen. To speak to the generation or transportation advancement issue, an emblematic coding was utilized for every part (chromosome) of the populace. In this example, client orders are spoken to by discrete whole numbers. Subsequently, every individual from the populace is a change of client orders.

Planning kinds of uses have underlined the organization of genetic calculations as a basic or single algorithmic procedure. This has some constraint level and has multifaceted nature of the issues fathomed and their prosperity. Ongoing examination distributions have built up the affectability of genetic calculations to the underlying populace. At the point when the underlying populace is arbitrarily produced, genetic calculations are appeared to be less proficient than the strengthening composes calculations, however, superior to the heuristic strategies alone.

Genetic algorithms depend on a likeness with the genetic structure and exercises of chromosomes inside a populace of individual. This new age contains a higher extent of the attributes controlled by the great individuals from the past age. Along these lines, over numerous ages, great qualities are spread all through the populace.

By favoring the mating of the more fit individual, the most encouraging zones of the pursuit space are investigated.

(i) **Initialization**: First and foremost numerous individual arrangements are haphazardly created to frame an underlying populace. The populace estimate relies upon the idea of the issue, yet commonly contains a few hundreds or thousands of conceivable arrangements. By custom, the populace is created haphazardly, covering the whole scope of conceivable arrangements. Every so often, the arrangements might be seeded in territories where ideal arrangements are probably going to be found.

(ii) **Fitness function**: A fitness function must be produced for every issue to be understood. For some issues, especially work advancement, and the fitness capacity ought to just gage the estimation of the capacity. Fitness function is a planned capacity that estimates the decency of an answer. It ought to be composed in the manner in which that better arrangements will have a higher fitness function an incentive than more regrettable arrangements. The fitness function assumes a noteworthy job in the determination procedure.

(iii) **Selection**: Good individual will most likely be chosen a few times in an age, and poor ones may not be incorporated. Distinctive kinds of determinations have been proposed to choose individual from a populace. The ordinarily utilized technique is roulette wheel test choice. The primary strategy of this is better individual get higher opportunity to take an interest to get the fitness solution. For that, dole out to every individual a piece of the roulette wheel at that point; spin the wheel n times to choose an individual.

In hybrid two individual genetic operators are browsed, the populace utilizing the selection operator and that joins a hybrid site along the bit strings is arbitrarily picked. At that point, the estimations of the two strings are traded up to certain point, and new offsprings are made. The two new offsprings made from this mating are put into the up and coming age of the populace. By recombining parts of good individual, this procedure is probably going to make far superior individual. Mutation is a genetic operator which has some low probability, and a bit of the new individual have a portion of their bits flipped. The reason for this is to keep up decent variety inside the populace and restrain untimely combination. Change alone prompts a subjective walk around the request space. Mutation is creating new offsprings from single parent and keeping up the assorted variety of the individual. All chromosomes are assessed to perceive how fit they are as arrangements. Individual with the most astounding fitness are constantly duplicated into the individual to come. Assessment positions the individual by some fitness measure that relates with the nature of the individual arrangements. The fitness calculation decides the reinsertion plans are global reinsertion for all populace-based choice calculation and nearby reinsertion for neighborhood choice. Distinctive plans of global reinsertions are produce the same number of offspring as parents and supplant all parents by the offsprings, deliver less offspring than guardians and supplant parents consistently at irregular, create less offsprings than parents, and it will supplant the most exceedingly bad parents and create more offspring than required for reinsertion and reinsert just the best offsprings. The age creation process is rehashed till a specific end condition has been come to. The normal ending conditions are an answer is discovered that fulfills least criteria, settled number of ages achieved, distributed calculation time or cash came to.

1.2 Cuckoo Search Algorithm

Cuckoo search is recently settled developmental optimization instrument which is absolutely reliant upon the rearing conduct of the Cuckoo. Like GA and PSO, CS [8–11] is likewise occupant focused calculation is presented in 2009 by Yang and Deb. This calculation is really endless supply of a few Cuckoos breed which is blend of some flight systems. It implies that Cuckoos can fly in the better places for finding the nourishment resource for their youngsters and for claim. Subsequent to finding the resource of sustenance, they can store or keep those resources in their own homes. Prior to the activity of finding the nourishment resource, the Cuckoo can leave their kids in their homes. Cuckoos leave their eggs in some area homes for advancement of security. Amid the seeking of nourishment, if the proprietor of the area is returned back to their own homes and found that it is not their own youngster or eggs then they may toss those eggs outside the homes.

Contribution of paper includes mainly:

- Showing hybrid GA + CS algorithm
- Minimizing makespan for the job scheduling problem.

2 Proposed Algorithm

To resolve job scheduling issue by using hybrid GA + CS algorithm is suggested by us. Principal purpose is to reduce makespan. GA + CS processes are fused together to make a hybrid procedure. GA often falls in local optima. For avoiding it, CS is used. Modified computation allocates jobs precisely than GA. Comparative outcome is also shown for the scheduling issue. For given issue, an arrangement of m machines and an arrangement of n jobs are given. Each issue comprises of an arrangement of tasks that must be handled in a predetermined grouping. Every task must be handled on an unmistakable machine and has a preparing time which is deterministically known. In single objective optimization, point is to discover a schedule that limits general culmination time known as makespan.

Principal purpose of the investigation:

- Algorithm ought not to ambush in local minima.
- Algorithm ought to apply local search quick.
- Algorithm ought to give superior outcomes to NP-hard issues.
- Algorithm ought to upgrade issue heuristically.

- To discover global best solution fast.

Stages of procedure are as follows:

Scheduling procedure:

1. Provide values to parameters.
2. Provide quantities of task and resource.
3. Apply new algorithm for the tasks.
4. Compute end time of tasks.
5. Check task allocation done. Yes, go for stage 6 else to stage 3.
6. End.

Genetic algorithm:

1. Initialize current generation.
2. While (condition).
3. Apply limited search using Cuckoo search process.
4. Apply choice operator.
5. Apply boundary operator.
6. Apply operator for genetic diversity.
7. Stop.

Cuckoo search algorithm:

Step-1 Initialize the populations of n host birds nest;

Step-2 While (stopping criteria).

Step-3 Choosing Cuckoo arbitrarily to supplant with Mantegna algorithm;

Step-4 Fitness parameter calculated.

Step-5 Arbitrarily find new nest, like *j*.

Step-6 Compute quality parameter *Fj*.

Step-7 Compare fitness value of both cuckoos.

Step-8 Contrast fitness value for both and change cuckoo *j* by novel one.

Step-9 Shallow fitness value nest will be avoided and novel will be created.

Step-10 Go for the best one.

Step-11 Select current as best one for future.

Step-12 End while.

3 Experimental Setup and Result

For high-quality outcome, various computations have few control parameters. So suggested algorithm also has few control parameters. After, tremendous literature survey-related parameters are found and applied. These values are benchmark values and applicable for trials [8–11]. These are pc (crossover probability), pm (mutation probability), and pa range is 0–1. Alpha value set to 0.3–1.99.10–100 is taken for max population. After that same process is applied for hybrid algorithm. This process is applied for various iterations. Genetic and Cuckoo-based hybrid algorithm provided better results as we have checked it for various iterations. As this represents single objective optimization so makespan time is calculated as output.

As per this observation, various iterations are checked. Number of resources and tasks/jobs are fixed for every iteration, and genetic algorithm is first applied. By applying algorithm, for every job time duration is checked. This process is applied for various iterations as shown in Table 1.

These values are benchmark values and applicable for trials. These are pc (crossover probability), pm (mutation probability), and pa range is 0–1. Alpha value set to 0.3–1.99.10–100 is taken for max population. After that same process is applied for hybrid algorithm. This process is applied for various iterations. Genetic and Cuckoo-based hybrid algorithm provided better results as we have checked it for various iterations. As this represents single objective optimization so makespan time is calculated as output.

As per this observation, various iterations are checked. Number of resources and tasks/jobs are fixed for every iteration, and genetic algorithm is first applied. By applying algorithm, for every job, time duration is checked. This process is applied for various iterations.

Table 1 Results of various iterations on job shop scheduling problem

Number of iterations	Number of task	Number of resource	Genetic algorithm	Hybrid algorithm
10	[6, 16, 17]	[3, 5, 6]	0.6228 0.8552 0.8826	0.4360 0.1561 0.1782
50	[6, 16, 17]	[3, 5, 6]	2.3010 3.6358 3.7583	0.5644 0.2598 0.2690
150	[6, 16, 17]	[3, 5, 6]	6.3489 10.3785 10.8546	0.9741 0.6719 0.6998
250	[6, 16, 17]	[3, 5, 6]	10.5332 17.3612 17.9401	1.8521 1.5730 1.5256

4 Conclusion

As evident from the tables and graphs that changing the iterations is affecting the completion time and hybrid algorithm is performing well. The hybrid genetic-Cuckoo search algorithm and genetic algorithm are tested with taken inputs for task and resources, where number of resources are less than the tasks on JSP problems in finding optimal makespan values. Thus, it is found that performance of hybrid genetic-cuckoo search algorithm approach is effective in finding ideal solutions contrasted with that of different methodologies.

It is applied for single objective optimization improvement and examined by the hybrid metaheuristic methodologies.

References

1. Jafari Navimipour N, Sharifi Milani F (2015) Task scheduling in the cloud computing based on the cuckoo search algorithm. Int J Model Opt 5(1):44–47
2. Deng X (2013) A dynamic task scheduling strategy based on MVFSA in cloud computing environment. Appl Mech Mater 427–429:2596–2599
3. Tawfeek M, Elhady G (2016) Hybrid algorithm based on swarm intelligence techniques for dynamic tasks scheduling in cloud computing. Int J Intel Syst Appl 8(11):61–69
4. Riahi V, Kazemi M (2016) A new hybrid ant colony algorithm for scheduling of no-wait flowshop. Oper Res Int J 18(1):55–74
5. Slowik A, Kwasnicka H (2018) Nature inspired methods and their industry applications—swarm intelligence algorithms. IEEE Trans Industr Inf 14(3):1004–1015
6. Roy B, Sen A (2018) Meta-heuristic techniques to solve resource-constrained project scheduling problem. In: International conference on innovative computing and communications, pp 93–99
7. Rahmani Hosseinabadi A et al (2018) Extended genetic algorithm for solving open-shop scheduling problem. Soft Comput 23, 13:5099–5116
8. Kumar N et al (2021) Applications of new hybrid algorithm based on advanced cuckoo search and adaptive Gaussian quantum behaved particle swarm optimization in solving ordinary differential equations. Expert Syst Appl 172:114646
9. Cuong-Le T et al (2021) A novel version of Cuckoo search algorithm for solving optimization problems. Expert Syst Appl 186:115669
10. Alkhateeb F et al (2021) Discrete hybrid cuckoo search and simulated annealing algorithm for solving the job shop scheduling problem. J Supercomput
11. Madni S et al (2018) Multi-objective-oriented cuckoo search optimization-based resource scheduling algorithm for clouds. Arab J Sci Eng 44(4):3585–3602

Occlusion Problem in 3D Object Detection: A Review

Apurva Kandelkar, Isha Batra, Shabnam Sharma, and Arun Malik

Abstract In computer vision, 3D object detection has numerous applications such as robotics, augmented reality (AR), medical field, manufacturing industries, and safe autonomous driving. But the real-object detection may involve various problems such as noise, missing data, and occlusion problem. From past few years, the great progress in 3D object detection has been made. Object recognition and identification in occlusion remain a difficult challenge, despite recent breakthroughs in 3D object detection. The occlusion problem is one of the difficulties in object tracking. The paper highlights a number of research hurdles and open concerns that researchers must address.

Keywords Robotics · Augmented reality (AR) · 3D object · Occlusion problem

1 Introduction

Computer vision is an interdisciplinary branch of study that aims to interpret digital photos and videos at a high level. Deep learning, a subset of machine learning that employs algorithms to extract insights from data, is used in modern computer vision.

The original version of this chapter was revised: The second author's affiliation has been changed to "Lovely Professional University, Phagwara, Punjab 144001, India" and third author's affiliation has been changed to "CMR University, Bengaluru, Karnataka India". The correction to this chapter is available at https://doi.org/10.1007/978-981-19-2821-5_67.

A. Kandelkar (✉) · I. Batra · A. Malik
Lovely Professional University, Phagwara, Punjab 144001, India
e-mail: apurvakandelkar@gmail.com

I. Batra
e-mail: isha.17451@lpu.co.in

A. Malik
e-mail: arun.17442@lpu.co.in

S. Sharma
CMR University, Bengaluru, Karnataka, India

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_26

Computer vision is integrated into many areas of our life such as (1) healthcare industry: The variety of computer vision applications are used for medical image analysis, which helps to improve the medical diagnosis process significantly. (2) Autonomous Vehicles: computer vision and object detection application such as self-driving cars. (3) Insurance Claims Processing: Clients can be guided through the process of visually documenting a claim with the computer vision application. It can evaluate and send images to the necessary agents in real time. (4) Agriculture: Sophisticated weather analysis, planting, harvesting, weeding, monitoring, and plant health detection are all tasks that precision agriculture may assist with. (5) Defense and Security: Computer vision applications can help merchants identify consumers using images from security cameras.

Companies with strong security requirements, such as banks or casinos, can profit from computer vision applications. 3D object recognition, image processing, and visual object analysis are important and difficult tasks in the field of computer vision. The detection of 3D object is finding and identifying the object of certain class in image or video from real world. Object detection algorithms are extracting the features and learning algorithms to recognize object category instances. Despite the fact that 3D object detection and recognition algorithms have been extensively investigated over the last decade, there are still certain concerns, opportunities, and challenges to be addressed.

The computer vision performs three major tasks: object detection, object recognition, and object reconstruction. Object detection identifies the presence of an object in a scene while object recognition realizes the detected object in the image scene. Recognition deals with identification detection are where that object is available in the image. The recognized object is reconstructed or classification of objects in an image for the purpose of interpretation or identification. In short, the object recognition is what type of object is in the image and object using reconstruction methods. Detection is the process of determining the existence of something/object (e.g., determining whether an object exists in an image/video), whereas recognition is the process of determining an object's identity (e.g., recognizing a person/object). There are two kinds of characteristics: Local features represent the image patches (key locations in the image) of an object, whereas global features describe the image as a whole to generalize the complete object. Global features include contour representations, form descriptors, and texture features, whereas local features represent the texture in an image patch.

The phases of 3D object recognition consist of (i) Scene Acquisition—action of retrieving an image from scene, (ii) Scene Segmentation—Several objects in 3D scenes should be distinguished from the background. (iii) Key Point Extraction [1]—The unique point of an input image that is unaffected by rotation, scale, or distortion. (iv) Descriptor Extraction—After the extraction of key points, descriptors are computed on the key points, resulting in a description that is used to represent the input cloud. (v) Matching—After that, descriptors are matched to compare two point clouds.

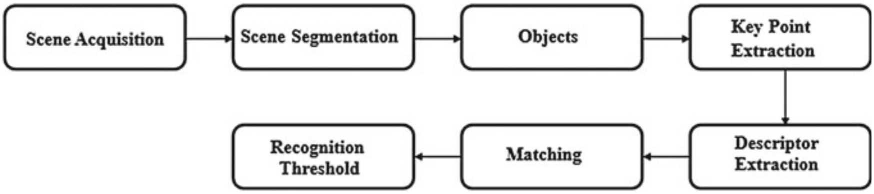


Fig. 1 3D object recognition phases

(vi) Recognition Threshold—determines the most similar input clouds as shown in Fig. 1.

1.1 Challenges in 3D Object Detection

The challenges in 3D object detection task are

- Missing data: If the input objects are occluded and some data is lost as a result.
- Noise: There is noise in sensors. Point perturbations and outliers are examples of distinct types of noise. This means that within a specific radius surrounding the location where it is sampled, a point has a certain disturbance occurs. It may emerge in space at random locations (outliers).
- Rotation invariance: Classification should be unaffected by rotation and translation points.
- Occlusion: Occlusion occurs when two or more objects come too close together and appear to merge or join.

1.2 The Occlusion Problem

In object detection approaches, each and every object in input image is detected independently without taking into account the object's surroundings. But in real time, the objects are not isolated from each other. It is difficult task to handle discrete sampling, occlusion problem, cluttered scenes, and noise scanning. The challenge is to handle occlusion problem that occurs between the objects [2]. The partial occlusion can introduce noise into the classification process. This partial occlusion may result in the misclassification of the object. Figure 2 shows 3D object detection occlusion problem.

In Fig. 2, the 3D space the object in set O are volume is represented as U (can be the subsets of U). Any or some part of O is intersected by ray r. The ray of light is interrupted by the O object if r intersects the O. The occluded O object (fully or partially) from the viewpoint v, if ray r is blocked by the different another object or if all r (rays) are blocked by o and v. The O is said to be completely visible if object is not blocked by v and O. From viewpoint v, the object o is visible but because of

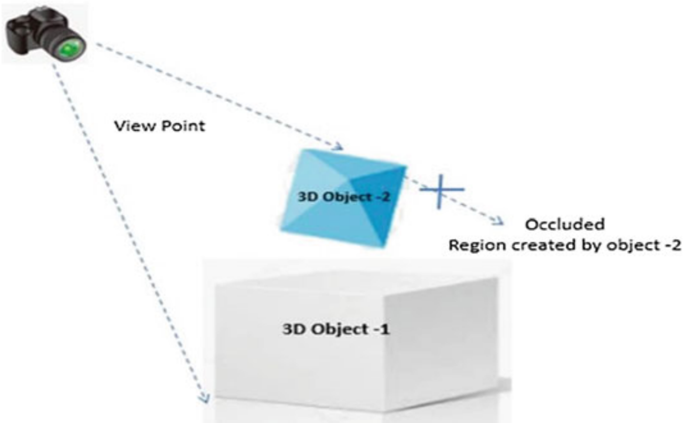


Fig. 2 Partial occlusion problem in 3D object detection

another object, the ray r is blocked between v and o, then the object is said to be partially occluded.

1.3 Motivation AND Contribution

The main objective of the survey is to provide comprehensive overview of several conventional and existing methodologies. This survey includes several important contributions:

1. The paper first describes the computer vision applications, 3D object detection challenges, and occlusion problem in 3D object detection.
2. Then, the paper describes the study based on various methodology and input circumstances used in occlusion detection problem.
3. This paper provides the analysis of various occlusion scenario and datasets used by respective methodologies.
4. Then, the paper identified the different application areas where occlusion detection technologies were applied.

The paper is organized as follows. Section 1 gives the introduction about computer vision, its applications, and 3D object detection phases. Then, it describes how occlusion problem occurs and it is handling 3D object detection. In Sect. 2, we present the related work on various methodologies used for handling occlusion problem for 3D object detection. Section 3 presents comparison of different methodologies used for 3D occlusion detection. Section 4 states comparison of performance evaluation methods. The last section concludes this paper and suggests the possible directions for the future works.

2 Review of Literature

The paper [2] proposed compositional convolutional neural networks (CompositionalNets) which detect partially occluded objects. The occluder is found, and the non-occluded regions are used to identify objects. The model is robust for partially occluded objects. It substitutes a differentiable compositional model for the DCNN's fully linked classification head, which can be trained from start to finish. The model is used on a variety of artificial occlusion levels (0, 20–40, 40–60, and 60–80% of the object are occluded) and occlusion kinds (o = natural, t = textured, objects w = white boxes, n = noise boxes). The average detection result is 75% on PASCAL3D + vehicles detection under occlusion.

In this paper, different approaches like histogram of oriented gradients (HOG), support vector machine (SVM), and convolutional neural network (CNN) are used for human detection. In this strategy, the human is completely exposed to complete occlusion, which is covered by other items in the environment, which utilized video in a noisy setting and in certain frames [3]. So result showed that for human tracking, CNN using KF is performing better performance throughout the video. This paper states that for real-world applications for complete and noisy occlusion videos, CNN works successfully throughout video. And for non-occlusion, the result provided by HOG-SVM is successful.

The paper [4] proposed k-means segmentation approach that successfully removes occlusion while 3D reconstruction of the images. It rapidly identifies the occluded region using adaptive k-means segmentation. The adaptive K-means segmentation method is used to identify the occluded region, and in second step, proposed algorithm is used to remove the occlusion.

In this paper [5], the DCNN and the compositional object model are combined to handle partial occlusion. In the proposed method, the DCNN is trained to handle the images classification and cluster the DCNN features into dictionary. The changes in 3D pose of the objects that is spatial activation patterns are recognized by the mixture of the compositional models. The model perform well for the non-occluded objects. The DCNN form features enable to represent the complex object in the natural scenes as it is difficult to achieve with related to the approaches of complex physical approaches such as illumination and local deformation.

The research [6] developed a unique spatio-temporal RGB-D video segmentation system that automatically segments and tracks objects in real time. To solve inconsistency under occlusion, the suggested solution employs SIFT flow and bilateral representation. The disadvantage of prior approaches, according to the article, is that they may lose pixel correspondence between frames under occlusion and fail to produce consistent and continuous segmentation results.

The region proposal network (RPN) is proposed in this paper [7], which shares full-image convolutional features with the detection network, allowing region recommendations to be produced for almost no cost. A RPN is a fully convolutional network that predicts object limits and scores at each point at the same time. The proposed method is extremely fast. The RPN enhances the quality of region proposals and as a result, the total object detection accuracy. The test-time as well as the marginal cost of computing suggestions are both low (e.g., 10 ms per image).

Each training and testing example has a collection of picture instances of an object and covers substantial changes in the object's appearance, and each training and testing example contains a set of image instances of an object and covers large variations in the object's appearance [8]. In this method, the occlusion can be detected by observing the changes of the dominant object depth values within the target bounding boxes.

The novel sparse voxel-graph attention network (SVGA-Net) was introduced in the publication [9] for 3D object detection from raw point clouds, which allows for a better local representation of the point feature and information fusion between the point and its neighbors. Sparse voxel-graph attention network (SVGA-Net) is basically a novel end-to-end trainable network which mainly contains voxel-graph module and sparse-to-dense regression module to achieve comparable 3D detection tasks from raw LiDAR data.

In this paper, the novel neural network method is proposed for generating a 3D-object point cloud model from a single-view image [10]. This structure allows 3D-ReConstnet to infer a 3D point cloud directly from 2D image features, eliminating feature propagation throughout the network and feature loss in multi-stage networks.

The methodology provided in paper [11] is designed to be utilized in an embedded system that uses a lightweight neural network to visually categories objects and also tracks them in 3D space using LiDAR data. The proposed system works for both classification as well as detection of the 3D object.

A variety of existing algorithms for object identification and tracking have been addressed in this study [12]. The author discusses the occlusion problem as an essential topic. This paper review different object detection and tracking under various level of occlusion. The article considers a variety of features, including depth data, geometry, textural, color feature metrics, speed, and so on.

Deep convolutional neural network (CNN) architecture is proposed in research [13] to localize semantic pieces in 3D and 2D space while inferring their visible states. The author employed a 3D object CAD model to generate large-scale synthetic data and simulate complicated occlusion combinations between objects. On public datasets, the suggested approach performs better on 2D and 3D landmark prediction even when occlusion and truncation are present.

The proposed method is used for object recognition, and pose estimation algorithm was implemented with Point Cloud Library (PCL) [14]. The data points of each object cluster are uniformly down-sampled for feature extraction using the uniform sampling (US) method, and the remaining points are regarded the feature points of each discovered item. This system not only recognizes an object in a complex image but also calculates the object's 3D position with reference to the camera.

The [15] paper suggested a new encoder that uses PointNets to build a representation of point clouds arranged in vertical columns (pillars), enabling for end-to-end learning by using 2D convolutional layers. In PointPillar, the speed and precision are excellent. The speed and accuracy are very good in PointPillar. It states that the

PointPiller is the best for faster speed, and also PointPillars offer the best architecture so far for 3D object detection from LiDAR.

VoxelNet is a universal 3D detection network that incorporates feature extraction and bounding box prediction into a single stage, end-to-end trainable deep network, according to paper [16]. This incorporates use of a new voxel feature encoding (VFE) layer that combines point-wise data with a locally aggregated feature to enable inter-point interaction within a voxel. The drawback is that performance is slow in real-time. The inference time for VoxelNet is 33 ms. Extending VoxelNet for joint LiDAR and image-based end-to-end 3D detection will increase detection and localization accuracy, according to future studies.

LaserNet is a computationally efficient approach for 3D object detection from LiDAR data introduced in [17] research. Using a fully convolutional network, our method predicts a multimodal distribution over 3D boxes for each point, and then effectively fuses these distributions to provide a forecast for each item. This method provides lower compared to other detector.

The model "RotationNet" developed in this paper is based on convolutional neural networks (CNNs), and it takes multiview photos of an object as input and estimates its pose and object category concurrently [18]. The advantage of this method is high accuracy in both object categorization and pose estimation. The main limitation of RotationNet is that each image should be observed from one of the pre-defined viewpoints.

In this paper, Canny edge detection algorithm is proposed [19]. The SLAM technology is used to ensure that the virtual model is placed over the 3D object in a stable manner. The median filtering is adopted in order to extract the contour of the object instead of Gaussian fuzzy. In Canny, the Gaussian filter is replaced by the median filter, which lowers edge information loss. According to the author, the proposed approach is capable of reliably recognizing 3D objects and displaying the virtual model above them in a stable manner.

The proposed method is using the concept of 3D mirroring; it introduces a new interaction metaphor for selecting obscured virtual objects with greater efficiency [20]. This work introduced 3D mirrored object selection, which has the advantage of allowing an obscured target object to be selected without having to change modes. As a result, this technique has quicker selection times and lower error rates.

This study [21] introduces a new end-to-end multiview fusion (MVF) approach that can effectively learn to employ complimentary information from both the front view and the birds-eye view. In addition to LiDAR, MV3D increases 3D object detection accuracy by merging CNN characteristics acquired from multiple angles (front view, birds-eye view, and camera view).

The author suggested a convolution neural network for 3D object class detection, its accuracy, and the effect of unfavorable conditions in real-world scenarios in this research [22].

In this paper, [22] the author proposed convolution neural network applied to 3D object class recognition, its accuracy, and the effect of adverse conditions in real-world scenes.

The author explains how to represent 3D data in order to feed it into the network. The result in paper shows that volumetric representation of 3D data plays important key role on recognition process and if proper representation is chosen the convolution neural network (CNN) is robust to noise and occlusion.

The paper [23] proposed the local surface descriptor method which works efficiently for 3D object detection under occlusion and in cluttered scene. The unique key and local frame are constructed for the key points and its local neighbor points. The proposed method efficiently recognizes and segmenting the target from cluttered scene. The object recognition rate is very high in the proposed algorithm.

In the paper [24] for 3D object detection from raw point cloud, there are two stages are proposed in the first stage bottom up 3D proposal generation is used, and in second stage, the result is refined in canonical coordinates to get final result of detection. The both stages are used for accurate box refinement and confidence prediction. This method is very correct and robust 3D detection performance, and it is very efficient and effective for 3D bounding box regression. The paper [8] proposed the 3D object tracking method which covers large variation in object appearance. The method used image sets and detection of depth image occlusion. Every image is represented as second order statistics. The kernel is used and to avoid improper appearance of model updating during occlusion depth information is used.

This method can detect occlusion problem by observing change of dominant object depth values using targeted bounding box.

The method proposed in [25] recognizes object and pose estimation using algorithm that was implemented with Point Cloud Library (PCL) and support vector machine (SVM). It not only recognizes objects in complex scenes, but it also reliably predicts the object's 3D posture in relation to the camera. The proposed algorithm is two stage machine learning algorithm that speeds up feature matching process in condition of rapidly increasing features.

The proposed method in [26] is OrthographicNet, which consists of a convolutional neural network (CNN)-based model, used for 3D object recognition in open-ended domains. For learning about the instances of a category, this method used an instance-based learning and recognition (IBL) approach, in which a category is represented simply by a set of known examples. The benefit of instance-based learning is that it can quickly adapt an object category model to the previously unseen instances by either storing new instances or ignoring old ones.

The paper [1] proposed two algorithms for 3D selective search algorithm 1 for the 3D selective search algorithm 2 for the object pruning. The advantage is that the algorithm no performance loss because of 3D SS and object pruning. The testing time reduced to 50% instead of scanning possible location and orientation in 3D space. So overall process is speed up.

The [27] paper proposed 3D dynamic voxel (3DV) method for 3D action representation. The method transfer 3DV into point set form and the input to the system is PointNet++. The method is end-to-end learning manner for action recognition in 3D action. The proposed method has the advantage of extracting from all temporal splits and the entire depth video at the same time.

The proposed method in [28] proposed the CNN-based ellipse detector called as ellipse R-CNN to infer and represents the occluded objects as ellipse. The paper states that traditional CNN is not suitable for elliptical fitting. For elliptical object detection, the suggested method is a robust and compact ellipse regression based on mask R-CNN architecture. To compute the final detection score for better occlusion management, the enhanced region for regression stage and incorporate the U-Net structure for learning varied occlusion patterns. The suggested technique outperforms the competition on both synthetic and actual datasets of occluded and clustered elliptical objects. To generalize both occluded and un-occluded instances, the proposed method is robust ellipse regression.

In an urban street setting, the proposed technique in [29] leverages mobile laser scanning (MLS) point cloud data for 3D object retrieval and completeness evaluation. This technique employs a PointRCNN-based 3D object detection network. It uses SO-Net to detect incomplete 3D shape retrieval, and the benefit is that it provides completeness evaluation criteria for discovered objects. Table 1 shows the summary of different methodologies and their performance used for 3D object detection and occlusion problem.

3 Performance Evaluation

By considering various studies discussed in the previous section, the summary analysis is provided which is based on various input environment used (i.e., static input, dynamic input, real-time input, 2D input, and 3D input). Figure 3 shows summary analysis of input environment used by various 3D occlusion detection studies. Hence, analysis factor includes the different input environment used by various methodologies.

The above summary analysis concludes that the type of input environment selected by the various researchers included the 3D inputs as 34%, static input as 31%, 2D input 16% and real-time input, 11% of the dynamic input, and only 8% of the real-time input. From the above analysis, it can be observed that there are less number of the work is implemented upon the real-time inputs. But real-time input processing has many advantages.

4 Conclusion and Future Scope

This paper surveyed different approaches to overcome the problem of partial occlusion in 3D object detection. To deal with challenges in 3D object detection, depth information is very important parameter as compared to 2D objects detection. The paper first introduces the background knowledge of 3D object detection, challenges occurred during 3D object detection and occlusion problem. Then, this paper summarizes and analyzes the various approaches proposed and applied on various datasets

Table 1 Summary of different methodologies and their performance used for 3D object detection and occlusion problem

Article	Studies	Occlusion scenario used	Methodologies used	Performance
[2]	Kortylewski (2020)	Synthetically occluded and real-time occlusion	Compositional convolutional neural networks (CompositionalNets)	– The model is robust for detection of partially occluded objects – The model is used on a variety of artificial occlusion levels (0, 20–40, 40–60, and 60–80% of the object are occluded) and occlusion kinds (o = natural, t = textured, objects w = white boxes, n = noise boxes) – The average detection result is 75% on PASCAL3D + vehicles detection under occlusion
[3]	Fatih Aslan et al. (2020)	Real-time occlusion	HOG-SVM and CNN method	– This research proposed different approaches like histogram of oriented gradients (HOG), support vector machine (SVM), and convolutional neural network (CNN) are used for human detection – For real-world applications for noisy and the fully occlusion videos, CNN works successfully throughout video

(continued)

Table 1 (continued)

Article	Studies	Occlusion scenario used	Methodologies used	Performance
[5]	Kortylewski et al. (2020)	Synthetically occluded and real-time occlusion	Deep convolutional neural networks (DCNNs) and compositional object models	– Deep convolutional neural networks (DCNNs) and compositional model to retain best approach that is robust to partial occlusion
[6]	Xie (2017)	Real-time occlusion	spatio-temporal RGB-D video segmentation framework	– This paper proposed an effective system for segmenting and tracking objects in RGB-D video under occlusion
[7]	Ren et al. (2015)	–	Region proposal network (RPN) and fast R-CNN	– The speed of the proposed is high – The RPN improves the quality of region proposals, which improves total object detection accuracy – The test-time as well as the marginal cost of computing suggestions are both low (e.g., 10 ms per image)
[8]	Chen et al. (2015)	Real-time occlusion	Novel 3D object tracking method using image sets and depth-based occlusion detection	– The paper proposed occlusion that can be detected by looking at changes in the dominating object depth values within the target bounding boxes
[9]	He et al. (2020)	Real-time occlusion	Novel sparse voxel-graph attention network (SVGA-Net)	– End-to-end trainable 3D object detection network from point clouds – Voxel-graph network is used to construct the local complete graph

(continued)

Table 1 (continued)

Article	Studies	Occlusion scenario used	Methodologies used	Performance
[10]	Li et al. (2020)	–	3D-ReConstnet: a single-view 3D- object point cloud reconstruction network	– Its innovative neural network technology converts a single-view image into a 3D-object point cloud representation

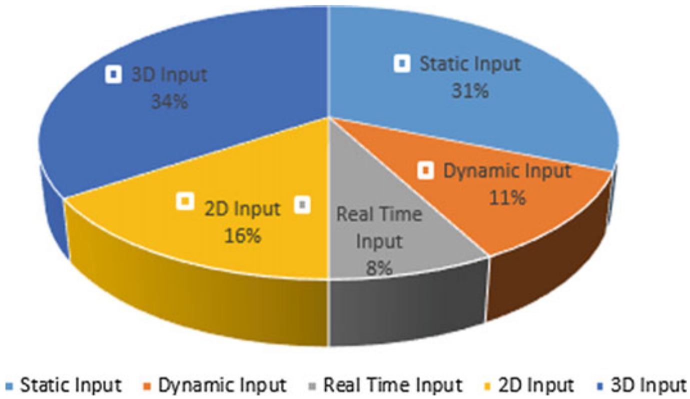


Fig. 3 Summary analysis of input environment used in occlusion detection studies

for handling occlusion problems in 3D object detection. Various ways for dealing with the occlusion problem for 3D object detection have been proposed, according to the overall literatures and their respective inferences. In future, further work is needed to be done for handling with occlusion problem in real-life circumstances. Future work will need working in a variety of environments, including dynamic environments. The future work needs to work on the dynamic environment. The dynamic environment also has great importance. This research will also benefit researchers to focus their efforts on the occlusion problem challenge for three-dimensional object detection.

References

1. Liu J, Chen H, Li J (2018) Faster 3D object detection in RGB-D image using 3D selective search and object pruning. In: 2018 Chinese control and decision conference (CCDC). IEEE, pp 4862–4866
2. Kortylewski A, Liu Q, Wang A, Sun Y, Yuille A (2020) Compositional convolutional neural networks: a robust and interpretable model for object recognition under occlusion. Int J Comput Vis 1–25

3. Aslan MF, Durdu A, Sabanci K, Mutluer MA (2020) CNN and HOG based comparison study for complete occlusion handling in human tracking. Measurement 158:107704
4. Priyanka A, Neelima N, Namithaa K, Raveena EP (2016) An innovative approach for occlusion elimination in 3D reconstruction. Procedia Comput Sci 87:111–115
5. Kortylewski A, Liu Q, Wang H, Zhang Z, Yuille A (2020) Combining compositional models and deep networks for robust object classification under occlusion. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision, pp 1333–1341
6. Xie Q, Remil O, Guo Y, Wang M, Wei M, Wang J (2017) Object detection and tracking under occlusion for object-level RGB-D video segmentation. IEEE Trans Multimed 20(3):580–592
7. Ren S, He K, Girshick R, Sun J (2015) Faster r-cnn: towards real-time object detection with region proposal networks. arXiv:1506.01497
8. Chen Y, Shen Y, Liu X, Zhong B (2015) 3D object tracking via image sets and depth-based occlusion detection. Signal Process 112:146–153
9. He Q, Wang Z, Zeng H, Zeng Y, Liu S, Zeng B (2020) SVGA-Net: sparse voxel-graph attention network for 3D object detection from point clouds. arXiv:2006.04043
10. Li B, Zhang Y, Zhao B, Shao H (2020) 3D-reconstnet: a single-view 3D-object point cloud reconstruction network. IEEE Access 8:83782–83790
11. Sualeh M, Kim GW (2020) Visual-LiDAR based 3D object detection and tracking for embedded systems. IEEE Access 8:156285–156298
12. Anuj L, Krishna MG (2017) Multiple camera based multiple object tracking under occlusion: a survey. In: 2017 international conference on innovative mechanisms for industry applications (ICIMIA). IEEE, pp 432–437
13. Li C, Zeeshan Zia M, Tran QH, Yu X, Hager GD, Chandraker M (2017) Deep supervision with shape concepts for occlusion-aware 3d object parsing. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5465–5474
14. Tsai CY, Tsai SH (2018) Simultaneous 3D object recognition and pose estimation based on RGB-D images. IEEE Access 6:28859–28869
15. Lang AH, Vora S, Caesar H, Zhou L, Yang J, Beijbom O (2019) Pointpillars: fast encoders for object detection from point clouds. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 12697–12705
16. Zhou Y, Tuzel O (2018) Voxelnet: end-to-end learning for point cloud based 3d object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4490–4499
17. Meyer GP, Laddha A, Kee E, Vallespi-Gonzalez C, Wellington CK (2019) Lasernet: an efficient probabilistic 3d object detector for autonomous driving. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 12677–12686
18. Kanezaki A, Matsushita Y, Nishida Y (2018) Rotationnet: joint object categorization and pose estimation using multiviews from unsupervised viewpoints. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5010–5019
19. Gao T, Yang Z (2020) 3D object recognition method based on improved canny edge detection algorithm in augmented reality. In: 2020 IEEE 5th international conference on image, vision and computing (ICIVC). IEEE, pp 19–23
20. Lee JJ, Park JM (2020) 3D mirrored object selection for occluded objects in virtual environments. IEEE Access 8:200259–200274
21. Zhou Y, Sun P, Zhang Y, Anguelov D, Gao J, Ouyang T, Guo J, Ngiam J, Vasudevan V (2020) End-to-end multi-view fusion for 3d object detection in lidar point clouds. In: Conference on robot learning. PMLR, pp 923–932
22. Garcia-Garcia A, Garcia-Rodriguez J, Orts-Escolano S, Oprea S, Gomez-Donoso F, Cazorla M (2017) A study of the effect of noise and occlusion on the accuracy of convolutional neural networks applied to 3D object recognition. Comput Vis Image Underst 164:124–134
23. Guo W, Hu W, Liu C, Lu T (2019) 3D object recognition from cluttered and occluded scenes with a compact local feature. Mach Vis Appl 30(4):763–783

24. Shi S, Wang X, Li H (2019) Pointrcnn: 3d object proposal generation and detection from point cloud. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 770–779
25. Luo W, Yang B, Urtasun R (2018) Fast and furious: real time end-to-end 3d detection, tracking and motion forecasting with a single convolutional net. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3569–3577
26. Kasaei SH (2020) OrthographicNet: a deep transfer learning approach for 3D object recognition in open-ended domains. IEEE/ASME Trans Mechatr
27. Wang Y, Xiao Y, Xiong F, Jiang W, Cao Z, Zhou JT, Yuan J (2020) 3DV: 3D dynamic voxel for action recognition in depth video. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 511–520
28. Dong W, Roy P, Peng C, Isler V (2021) Ellipse r-cnn: learning to infer elliptical object from clustering and occlusion. IEEE Trans Image Process 30:2193–2206
29. Guo Y, Wen C, Sun X, Wang C, Li J (2019) Partial 3D object retrieval and completeness evaluation for urban street scene. In: IGARSS 2019–2019 IEEE international geoscience and remote sensing symposium. IEEE, pp 1252–1255

A Survey: Lightweight Cryptography Study for Healthcare Devices and Applications Within the Internet of Things

Sadoon Hussein and Ahmed Sami

Abstract The Internet of things (IoT) is growing more prevalent and popular in recent years as a result of interconnected entities that allow millions of gadgets to talk with one another. We can call IoT an evolution of the Internet, and it has gotten. In recent years, there has been a lot of interest from researchers. The IoT is significant because it allows many low-resource and restricted devices to communicate, calculate, and process many operations and make decisions in the communication network. With each technological improvement, the creation of intelligent systems with high communication and data collection capabilities becomes achievable, opening up new opportunities for a wide range of IoT applications, notably healthcare systems. On the other hand, everything is useful and is not without problems and challenges. There are several difficulties to deploying IoT in the real world, ranging from tiny sensors to servers, such as interoperability, portability, accessibility, privacy, and information security. In this article, we present a complete survey of IoT technologies, processes, statistics, and success stories applied to healthcare, with an emphasis on the security threats and needs of IoT cryptography, technology, and IoT device trends.

Keywords Lightweight cryptography · Healthcare · SPN · Feistel · GFN · IoT

1 Introduction

The IoT is a huge number of devices or objects that communicate with each other over the Internet to enable the user to monitor, analyze, and remotely control them. In recent years, the growth of IoT has dramatically occupied our lives in many areas such as city, hospitals, agriculture, environment, roads, and homes. The increasing

S. Hussein (✉)
Department of Physic, College of Science, University of Mosul, Mosul, Iraq
e-mail: sadosbio113@uomosul.edu.iq

A. Sami
Department of Computer Science, College of Computer Science and Mathematics, University of Mosul, Mosul, Iraq
e-mail: ahmed.s.nori@uomosul.edu.iq

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_27

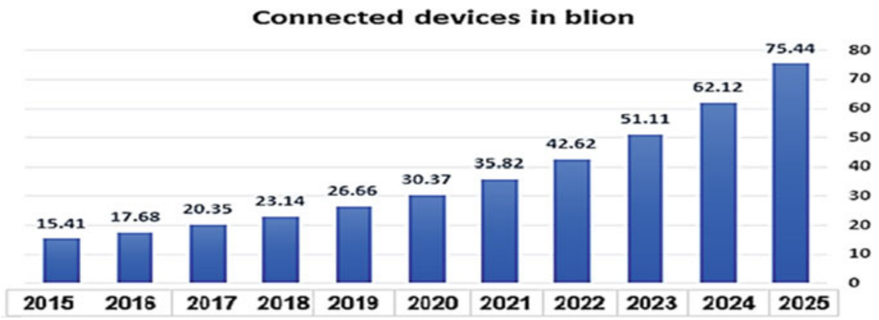


Fig. 1 Number of connected devices in IoT

of the IoT makes it the next future of technology in human history. The IoT device is typically outfitted with a variety of sensors and actuators that collect a large amount of data and transfer it over the Internet for monitoring, analysis, control, and drawing various conclusions [1]. The majority of this information is current and assists us in making informed decisions about various aspects of services providing in various fields. This Internet-based raw data, however, must be transferred securely [2]. Hackers can easily gain access to the corporate network and personal assets if IoT devices are left unprotected or vulnerable. Due to limited storage, processing capacity, and computing power, traditional security measures are insufficient and cannot be applied to IoT technologies. Furthermore, when a variety of devices are interconnected, scalability and heterogeneity issues arise. The IoT branch has been steadily developed over the last decade. There are about 14.2 billion linked devices in use. According to Gartner, Inc., this number is predicted to reach 25 billion devices by 2021. From 2015 to 2025, Fig. 1 depicts the global number of linked devices [3]. The following are the three IoT components that enable consistent computing anywhere: **I. Sensors**, actuators, and embedded communication hardware are all part of the hardware. **II. Middleware**- is a set of data analytics tools that include on-demand storage and compute **III. Presentation tools**- make it simple to comprehend, visualize, and interpret information. These tools are available on a variety of platforms and can be used by anyone [4]. Consumers of such technologies are expected to gain greatly from all of these advanced breakthroughs. Patients with serious conditions will be able to consult with their doctors and manage their condition while sitting at home thanks to the use of IoT in healthcare devices. Consumers will be able to turn on the lights, air conditioning, and television, as well as heat up their dinner, using home automation systems [5].

2 Healthcare and IoT

Healthcare systems are changing from time to other, and arguably the most notable of these changes was the world problem that occurred in the year 2020 with the advent of the new COVID-19 outbreak, in which the world's population was obliged to practice distancing and stay at home [6, 7]. As international leaders examine strategies to improve access to technology-assisted healthcare in response to (and in the aftermath of) the current COVID-19 issue, it is becoming progressively vital to understand how established and developing IoT technologies may support health systems in providing safe and effective treatment in a complementary or alternative way amid illness tidal waves [8]. Jaimon T. Kelly and et al. present in their research that perspective of present healthcare technology, how IoT devices are boosting health service delivery, and how IoT technologies will impact global healthcare in the coming decade. The IoT disruption in healthcare can lead to enhanced access and equitable primary, secondary, and tertiary smart healthcare, according to this opinion paper [9]. We will start the discussion on the IoT with a brief description of IoT and some examples of past projects in that area.

2.1 IoT

Sensors, actuators, and other IoT devices that have been integrated with physical equipment to communicate and supervise data using various protocols and modes of communication such as IEEE 802.11 (Wi-Fi), Bluetooth, Zigbee, and others. The sensors use in different fields, and for different purpose, for example, it can use in healthcare applications as well as in environmental information. The IoT can be used in healthcare applications to collect vital physiological data from patients' bodies, such as temperature, blood pressure, ECG, EEG, and so on. In this case, it may be wearable on the human body or embedded. While it can use to record important information such as humidity, temperature, date, and time in an environmental information. The data recorded by the sensors can help to make accurate and targeted conclusions about patients' health conditions and environmental conditions such as weather and climate. Through this data, it is possible to predict what will happen in future and what is the appropriate decision that can be taken regarding each case or event. The dealing with this data such as processing, storage, accessibility, and transmitted plays an important role IoT system [10]. Sharing this data with the concerned persons and related parties through the cloud/server allows for solving problems and rapid intervention in emergency cases if necessary. Most IoT systems have a graphical user interface that acts as a control panel for different service providers and decision makers for different situations and controls the user and displays data and states. There is a large number of researches in the literature presented in IoT systems, and most of these researches talk about the role of IoT in healthcare monitoring, control, security, privacy, and environmental predictions [11]. The great achievements that have been provided in the field of healthcare and other fields are the best evidence and witness to the effectiveness of the IoT and its appropriate future in many areas. In order to employ the IoT in healthcare systems, a number of countries have adopted special technologies, policies, and laws for this matter, which have transformed the healthcare sector into a more useful field to explore [12]. The motivation of this review is to provide a summary of the latest research and studies in healthcare systems based on the IoT and the security issues associated with them, also to give a systematic review of possible technologies, applications, and services.

2.1.1 Comparison with Related Reviews

Scopus publications between Jan 2012–Oct 2020 were gathered, yielding 472 review articles classified as "review". These articles were then expanded upon. as shown in Fig. 2 [13].

2.2 IoT Healthcare Architecture

The application perception layer and the network layer are the three critical layers of the IoT architecture in healthcare delivery. We will not go into detail about these layers; nevertheless, in the next parts, we will provide a concept of connected health repercussions [14]. Perception Layer: The core of IoT is sensing systems that collect data, as well as perception and identification technology. Network Layer: At the network level of IoT technologies, data exchange and storage between items can occur at low, medium, and high frequencies, with the latter being the primary

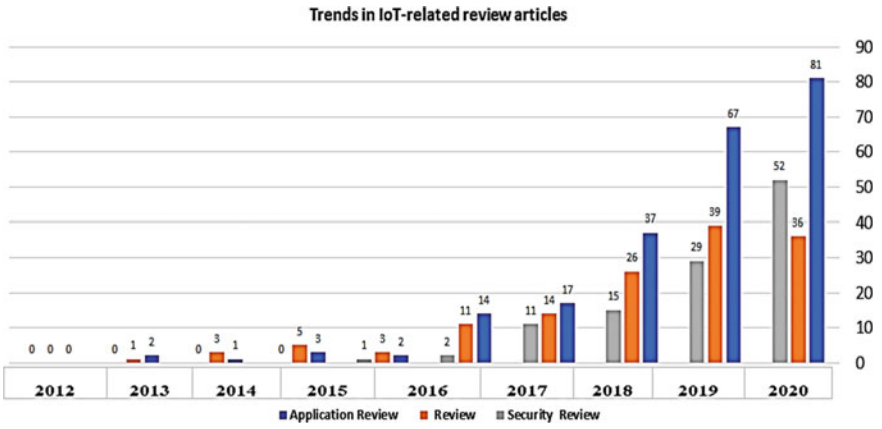


Fig. 2 Trends in IoT-related review article

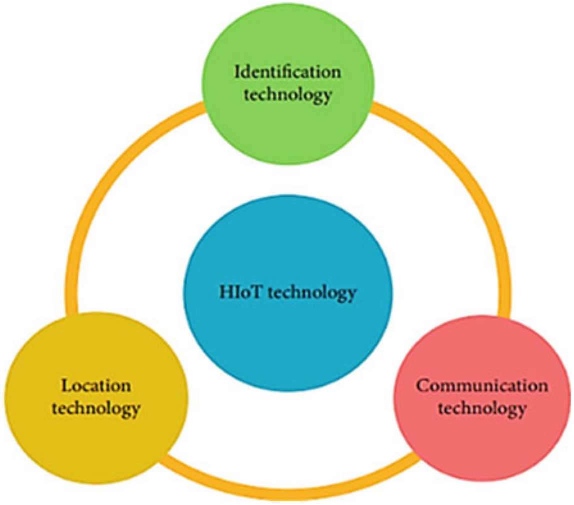


Fig. 3 Classification of IoT technology

emphasis of IoT. Application Layer: The function of the application layer is to interpret and apply data and is responsible for providing application specific services to the user [14].

2.3 Healthcare IoT Technologies (HIoT)

The employment of a specialized technology that can expand the capacity of the IoT system is what makes the technologies used to construct HIoT so important. To integrate diverse healthcare apps with the IoT system, a number of new technologies have been used. Identification, communication, and location are the three categories in which these technologies might be classed see Fig. 3 [15].

2.4 HIoT Services and Applications

Recent advancements in IoT technology have enabled medical devices to do real-time analyzes previously unavailable to clinicians. It also aided healthcare centers in reaching out to a larger number of individuals at once and providing high-quality care at a reasonable cost.

The IoT's growing influence has aided the development of HIoT applications such as disease diagnosis, personal care for children and the elderly, health and fitness management, and chronic disease monitoring. These applications have been separated into two fundamental categories, namely services and applications, to help you understand them better. The first category contains concepts that are employed during the construction of an HIoT device, while the second category covers healthcare applications that are used in either diagnosing a specific health problem or measuring health standards [12].

2.5 Future Trends of IoT in Healthcare Market

The rising usage of healthcare information systems, the growth of BigData in healthcare, improved device accuracy and connectivity, and the expanding penetration of connected devices in healthcare are all driving the IoT industry. IoT technology has the potential to alter traditional paper-based healthcare therapies by increasing access to real-time patient data and remote patient monitoring. The emergence of digital healthcare technology has addressed the pressing need for enhanced diagnostics and personalized treatment options. It also serves as a fitness and wellness tracker for athletes, as well as a dosage reminder for patients, in addition to providing remote patient monitoring to clinicians. The IoT healthcare market is in high demand due to the effective deployment of IoT in remote monitoring of diabetic and asthma patients, as well as the increased penetration of fitness and wellness gadgets. Improvements in the quality of people's everyday health and living activities, as well as BigData analytics to support and encourage such research efforts, are some of the factors driving market growth as IoT-based healthcare systems play a significant role in the expansion of medical information systems. Patients must be tracked, traced, and monitored in order to improve the healthcare system [14].

3 Security Threats for IoT, Countermeasures Based on Cryptography

The key difference between IoT and traditional systems in terms of security is that any devices that misuse real-world data collection will be diverted as a target for attackers. Consider a mechanical plant where IoT can be utilized to boost production and ease maintenance by coordinating and collecting data from the enormous number of sensors installed across the facility. If wrong information is received in the server as a result of cyber-attacks, it leads to incorrect analysis, which leads to incorrect management, which leads to massive damage in a large mechanical factory. In terms of management and manufacturing process, measurement and control data are more confidential, and preventing leakage is a critical goal from a competitiveness standpoint. As a result, while deploying any IoT-based systems, it is critical to evaluate the impact of security threats [16]. The three key reasons for cyber attackers' ease of assault against IoT are as follows. The first is that the IoT lacks a supervisory or intelligent system for detecting and identifying intruders. Second, because the Internet of things is based on wireless technology, snooping is rather easy. Finally, IoT devices are willing to accept limited capacity in terms of both power consumption and computational power. As a result, standard security procedures that are computationally expensive to create and install will impede the performance of power-controlled devices. Hence cryptographic methods play a critical role in protecting safe mode in order to transmit data to IoT without any data leakage. Unauthorized access to data transmitted over the Internet poses a risk of tampering with control signals or issuing illegal directives, resulting in anomalous operations.

3.1 Lightweight Cryptography

The basic goal of encryption is to keep information private so that only the sender and receiver can recognize and use it, and no other person or intruder can see or use it. The majority of current security standards algorithms have a greater gate count and power consumption. Therefore, these algorithms are almost impossible to implement in IoT devices. Another process named "lightweight cryptography (LWC)" has been created to define more suitable security with less equipment utilization and more optimized results for IoT devices. One of the most critical components of a safe embedded system is lightweight encryption. The newest development algorithms in this subject, as well as the most recent applications in the field of lightweight symmetric key cryptography, will be presented in this part. For platforms like RFID, NWSN, field programmable gate array (FPGA), and others, lightweight encryption techniques make more sense. Furthermore, symmetric shapes and asymmetric numbers are separated from lightweight encryption techniques. Lightweight symmetric block ciphers are used in the most common form. The mass and current zeros are contained in symmetric zeros [17] end-to-end security in IoT systems necessitates the use of symmetric encryption. An encryption mechanism with a low-power consumption is also particularly crucial for low-power devices [18]. Lightweight application is a dual core base that allows lower power consumption of peripheral devices. Any of the characteristics, such as memory capacity, latency, power, and circuit size, can be optimized using a lightweight encryption algorithm base. The use of coding in detectors shows that deploying a knowledge keeper for privacy and reliability could be a good counter-threat strategy. Lightweight encryption as shown in Fig. 4 has the benefit of enabling a device to securely encrypt, even for nodes with limited resources [19, 20].

3.1.1 Lightweight Cryptography Requirements

These are implementation aspects which are necessary for lightweight coding.

- Size (circuit size, ROM/RAM sizes)
- Power

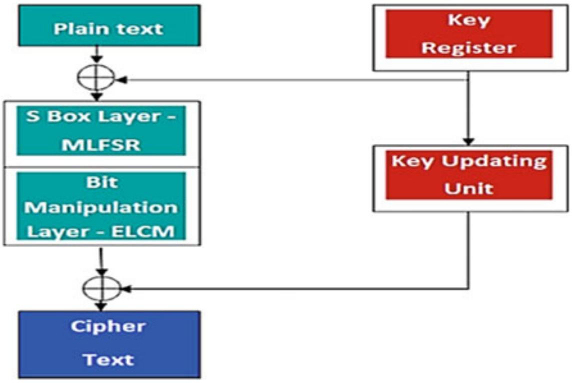


Fig.4 Block diagram of proposed E3LCM method

- Power consumption
- Processing speed (throughput, delay)

3.1.2 Good Lightweight Ciphers Features

A lot of factors must be considered while designing at good, memory efficient, resource-saving, durable, and lightweight encoder. The following are important characteristics of excellent and lightweight blade: (a) rich encryption standard, (b) less time to implement, (c) less complex, (d) high productivity, (e) solid structure, (f) consumes less power (energy consumption), (g) need to implement smaller devices, (h) requires less memory, executing programs (code size, RAM size), (i) strong resistance to linear and differential assaults. (j) Prevent possible advance attacks like zero correlation attack, Biclique attack, meet-in-the-middle attack, algebraic attacks, etc. [21].

3.2 Lightweight Cryptography for IoT Devices

Recently a many of software and hardware implementation of lightweight ciphers are designed for IoT applications with a focus on reducing cost (memory, processing power, physical area ,Gate Equivalents, energy consumption) and enhanced hardware and software performance (latency, throughput). Several of them focus specifically on security threats and are particularly concerned with implementation and/or performance cost [19, 22]. Furthermore, most IoT devices focus with real-time applications, making rapid and precise responses while maintaining basic security with limited resources a difficult task [23]. IoT device designers confront a number of threats and concerns, including battery capacity [20] and data security [24]. The foregoing concerns are well solved by its sub-specialty, lightweight cryptography, which

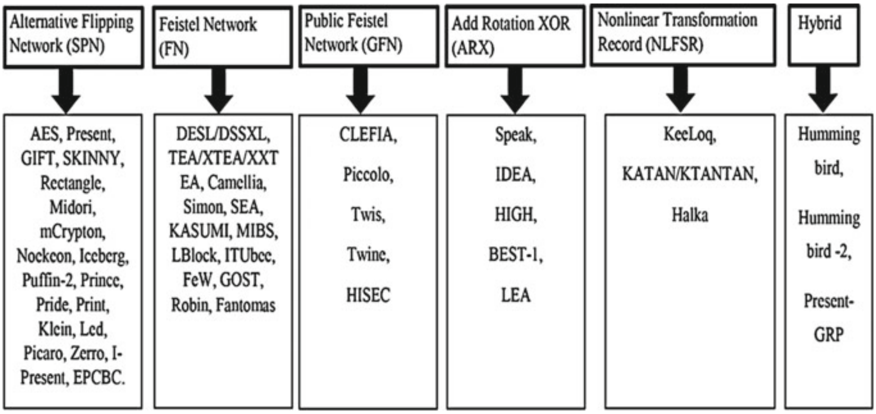


Fig. 5 Structure-wise categorization of algorithms

provides lightweight characteristics including tiny memory, low-power consumption, low-processing_power, and real-time response even on resource-constrained hardware [25]. Block ciphers are preferable over stream ciphers in IoT devices with limited resources for the reasons stated above. This section is about block ciphers, mostly symmetric lightweight block ciphers. It employs one of the structures as shown in Fig. 5.

SPN and TFN are the most popular structures among them because of their flexibility in implementation based on application requirements [26]. Although Feistel-structures are easy to include into low-medium power devices (because to the lack of circular function in half of the cases), they often require more circular function for safety reasons than SPN structures. When a choosing between fewer rounds of SPN work and greater Feistel job cycles with the same level of safety and equivalent energy costs is available, the SPN job may be the better option [27].

4 The Performance Metrics of Hardware and Software

Based on the structures mentions above, the performance of LWC algorithms might be assessed in terms of key size, block size, throughput, efficiency, and N of rounds, as well as hardware and software specific resource requirements. Tables 1 and 2 provide the performance metrics for both software and hardware.

Table 1 Existing lightweight block ciphers (software implementation)

No	Cipher	Year	Structure	KeyTsize (bits)	BlockTsize (bits)	Throughput@4 MHz’ (Kbps)	Efficiency (Kbps/KB)	No. of rounds
1	AES	2001	SPN	128/192/256	128	122	132.9	10/2/14
2	PRESENT	2007	SPN	80, 128	64	23.7	35.91	31
3	PRINCE	2012	SPN	128	64	70.8	63.9	12
4	LED	2011	SPN	80	64	7.28	3.36	32
5	DESL	2007	Feistel	56	64	30.6	9.88	16
6	LEA	2013	SPN with ARX-based S-boxes	128/192/256	128	34.5	53.24	24/28/32
7	SIMON	2013	Feistel	64, 72, 96, 128, 144, 192, 256	32, 48, 64, 96, 128	323	1900	32, 36, 42, 44, 52, 54, 68, 69, 72
8	LBLOCK	2011	Feistel SPN	80	64	13.48	13.81	32
9	PICCOLO	2011	GFN	80	64	11.93	12.35	25/31
10	GIFT [26]	2017	SPN	128	64/128	40.14	7.02	28/40
11	DLBCA [27]	2017	Feistel	80	32			15
12	Hybrid Symmetric [28]	2019	Block	64	128	High	Fast	32
13	Modified PRESENT [29]	2020	SPN, FN	64	80	High	Fast	25
14	MBCT [30]	2020	Block	256	256	High	Fast	32

Table 2 Existing lightweight block ciphers (hardware implementation)

No	Cipher	Year	Structure	KeyTsize (bits)	BlockTsize (bits)	Throughput @100 kHz (Kbps)	Efficiency (Kbps/KGE)	No. of rounds
1	AES	2011	SPN	128	128	56.64	23.6	10/12/14
2	PRESENT	2007	SPN	80	64	200	127.38	31
3	RECTANGLE	2015	SPN	80	64	246	167.68	25
4	PRINCE	2012	SPN	128	64	533,3	180.59	12
5	LED	2011	SPN	64	64	5.1	5.27	32
6	SIMON			96	48	15,8	20.7	32/52/72
7	LBLOCK	2011	Feistel + SPN	80	64	200	151.51	32
8	PICCOLO	2011	GFN	80	64	237.04	208.66	25/31
9	TWINE	2011	GFN	80	64	178	118.42	36
10	KATAN	2009	NLFSR	80	32	12.5	15.58	254
11	KTANTAN	2009	NLFSR	80	32	12.5	27.05	12/16/20

5 Security Analysis of Lightweight Algorithms

The optimal algorithm of LWC must preserve a proper balance among cost, performance, and security. To achieve any two of three can be optimized but to achieve all three is very difficult. So, to achieve these when design an algorithm of LWC must concentrate on less computing power requirement and less memory, resulting in less gate equivalent requirements along with low-power consumption without compromising strong security. Almost all of the recent surveys have been done to find out performance of any cryptographic algorithm based on the measured in terms of latency, key size, block size, throughput, efficiency, and number of rounds, power, energy consumption, and efficiency. Resources for hardware implementations can be expressed in the form of slices, gate area, or gate equivalents and efficiency.

6 Conclusion

Each lightweight cryptography algorithm used in the IoT has various advantages and disadvantages for ensuring security during information exchange in the IoT environment, according to the lightweight cryptography algorithms used in the IoT. Some algorithms require greater storage space but fewer computations, while others require the opposite. Many algorithms are low-power, low-cost, and low-computing-power; nonetheless, they do not demonstrate resistance. Because it has not been effectively handled, security is an essential concern in the IoT, not only now but in future, and it remains an important study area in the protection of the IoT. Future research should concentrate on reducing key size, employing more frequent dynamic keys, reducing block size, offering clearer rounds, and building simple key tables for lightweight block cipher development, according to our recommendations. Thus, one of the primary goals of developing lightweight stream ciphers is to reduce the size of the switch and to initialize the vector.

References

1. Hameed A, Alomary A (2019) Security issues in IoT: a survey. In: 2019 international conference on innovation and intelligence for informatics, computing, and technologies (3ICT), IEEE. https://doi.org/10.1109/3ICT.2019.8910320
2. Rana M, Mamun Q, Islam R (2020) Current lightweight cryptography protocols in smart city IoT networks: a survey. Available on: https://arxiv.org/ftp/arxiv/papers/2010/2010.00852.pdf
3. Mary Shamala L et al (2020) Lightweight cryptography algorithms for IoT enabled Networks: an overview. J Phys Conf Ser RASCC
4. Gubbi J, Buyya R, Marusic S, Palaniswami M (2013) IoT (IoT): a vision, architectural elements, and future directions. Futur Gener Comput Syst 29(7):1645–1660
5. Agrawal M, Zhou J, Chang D (2019) A survey on lightweight authenticated encryption and challenges for securing industrial IoT. Adv Sci Technol Secur Appl Springer
6. Torous J, Jän Myrick K, Rauseo-Ricupero N, Firth J (2020) Digital mental health and covid-19: using technology today to accelerate the curve on access and quality tomorrow. JMIR Ment Health 7(3):e18848 [FREE Full text]. [https://doi.org/10.2196/18848] [Medline: 32213476]
7. Fisk M, Livingstone A, Pit SW (2020) Telehealth in the context of covid-19: changing perspectives in Australia, the United Kingdom, and the United States. J Med Internet Res 22(6):e19264 [FREE Full text] [https://doi.org/10.2196/19264] [Medline: 32463377]
8. Ye Q, Zhou J, Wu H (2020) Using information technology to manage the covid-19 pandemic: development of a technical framework based on practical experience in China. JMIR Med Inform 8(6):e19515 [FREE Full text] [https://doi.org/10.2196/19515] [Medline: 32479411]
9. Kelly JT et al (2020) The IoT: impact and implications for healthcare delivery. J Med Internet Res 22(11):e20135
10. Peng H, Tian Y, Kurths J, Li L, Yang Y, Wang D (2017) Secure and energy-efficient data transmission system based on chaotic compressive sensing in body-to-body networks. IEEE Trans Biomed Circuits Syst 11(3):558–573
11. Gatouillat A, Badr Y, Massot B, Sejdic E (2018) Internet of medical things: a review of recent contributions dealing with cyber-physical systems in medicine. IEEE IoT J 5(5):3810–3822
12. Pradhan B et al (2021) IoT-based applications in healthcare devices. J Healthcare Eng
13. Lee JY, Lee J (2021) Current research trends in IoT security: a systematic mapping study. Hindawi, Mobile Inf Syst, Article ID 8847099, 25 p. https://doi.org/10.1155/2021/8847099
14. Sethi P, Sarangi S (2017) IoT: architectures, protocols, and applications. J Electric Computer Eng
15. Yuehong Y (2016) +e IoT in healthcare: an overview. J Ind Inform Integr 1:3–13
16. Prakasam P, Madheswaran M, Sujith KP et al (2021) An enhanced energy efficient lightweight cryptography method for various IoT devices. ICT Express. https://doi.org/10.1016/j.icte.2021.03.007
17. Shah A, Engineer M (2019) A survey of lightweight cryptographic algorithms for IoT-based applications. Springer Singapore Pte Ltd.
18. Prakasam P, Suresh Kumar TR, Velmurugan T, Nandakumar S (2019) Efficient power distribution model for IoT nodes driven by energy harvested from low power ambient RF signal. Microelectron J. https://doi.org/10.1016/j.mejo.2019.104665
19. Mohajerani R, Haeussler R, Nagpal F, Farahmand A, Abdulgadir J-P, Kaps, Gaj K (2020) FPGA benchmarking of round 2 candidates in the NIST lightweight cryptography standardization process: methodology, metrics, tools, and results. Available: https://eprint.iacr.org/2020/1207

20. Sheng Z, Yang S, Yu Y, Vasilakos A, Mccann J, Leung K (2013) A survey on the IETF protocol suite for the IoT: standards, challenges, and opportunities. IEEE Wirel Commun 20(6):91–98
21. Sehrawat D, Gill NS (2018) Lightweight block ciphers for IoT based applications: a review. Int J Appl Eng Res 13(5):2258–2270. ISSN 0973-4562
22. Biryukov A, Perrin LP (2017) State of the art in lightweight symmetric cryptography, Univ. Luxembourg Library, Esch-sur-Alzette, Luxembourg, Tech. Rep. 10993/31319. Available: https://orbilu.uni.lu/handle/10993/31319
23. Toshihiko O (2017) Lightweight cryptography applicable to various IoT devices. NEC Tech J 12(1):67–71
24. Singh S, Sharma PK, Moon SY, Park JH (2017) Advanced lightweight encryption algorithms for IoT devices: Survey, challenges and solutions. J Ambient Intel Hum Comput 4:1–18
25. McKay K, Bassham L, Turan MS, Mouha N (2017) Report on lightweight cryptography (Nistir8114). Gaithersburg, MD: NIST
26. Hatzivasilis G, Fysarakis K, P apaefstathiou I, Manifavas C (2018) A review of lightweight block ciphers. J Cryptogr Eng 8(2):141–184
27. Banik S, Bogdanov A, Isobe T, Shibutani K, Hiwatari H, Akishita T, Regazzoni F (2015) Midori: a block cipher for low energy. In: Proceedings 21st international conference theory applications cryptology information security, Part II. Springer, Berlin, pp 411–436. [Online]. Available: https://link.springer.com/chapter/https://doi.org/10.1007/978-3-662-48800-3_17
28. Banik S, Pandey SK, Peyrin T, Sasaki Y, Sim SM, Todo Y (2017) GIFT: a small PRESENT. In: International conference on cryptographic hardware and embedded systems. Springer, Cham, pp 321–345
29. AlDabbagh SSM (2017) Design 32-bit lightweight block cipher algorithm (DLBCA). Int J Comput Appl 166(8)
30. Kubba ZMJ, Hoomod HK (2019) A hybrid modified lightweight algorithm combined of two cryptography algorithms PRESENT and Salsa20 using chaotic system. In: 2019 international conference of computer and applied sciences (1st CAS2019), IEEE. https://doi.org/10.1109/CAS47993.2019.9075488

A Combinatory Novel Approach for Detection of Interested Area from Images

Gurbakash Phonsa and Gurpreet Singh

Abstract The main cause of the death as per today's scenario is brain sicknesses that effect the human's body activities along with brain cells. The detection of brain diseases is still a challenge due to many factors like user-friendly diagnoses interfaces, test accuracy, early detection stage, and many more. Keeping all this in mind, this paper proposes a hybrid approach for detection of brain tumor and eliminating all other replica factors. We use TDE hybridized with bisection mechanism that having history in illnesses detection. In our study, this approach shows a betterment in comparison with existing techniques with 8–10% margin. It is our understanding the use of objective function at classification and segmentation phase as well as enhancing contrast will show this improvement. These results are shown through parameters like MSE and PSNR in relationships of precision.

Keywords Genetic algorithm · Makespan · Cuckoo search

1 Introduction

The brain plays an important role in managing the different functions performed by our body that includes not only eating, movements, decision-making but countless functions in daily life. Now due to some defects, this operation if not performed accurately then can be termed as brain problem or brain sickness. This problem now started effecting the daily activities of the human being. Nowadays, there are numerous methods utilized in detection of these diseases. Mainly, the images are role players in all these detection techniques. These diagnosis methods are not very patient friendly as well are complex in nature. Thus, there is a scope for kind of techniques which detecting diseases in patient friendly method. There are many approaches

G. Phonsa (✉)
Mewar University, Gangarar, India
e-mail: phonsa@gmail.com

G. Singh
CSE, Punjab Institute of Technology, Rajpura (MRSPTU, Bathinda), India

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_28

which use image segmentation in detection process. This approach divides the image in different regions based on their parametric values. The boundaries of parameter specific region [1] used in object detection.

This approach integrated with many sections of informative parts that covers the entire image. In particular area, these pixels having similarities in terms of their representations like texture or color. Now, this machine computation approach gives the informative data as a result. That accepts the input in form of image and gives the output in the term of numerical values with respect to different regions in that input image. These output results in numerical values can be utilized in various forms of computation to enhance the diagnosis process. Now, the logical approaches can be deployed as per experiences to improve it further for betterment of diagnosis process.

2 Related Work

Tirpude and Welekar [1] images captured during medical tests are viewed properly only by means of medical image processing so it has develop into the most difficult fields of engineering and technology. Using this technology images taken during MRI reflects complete information about the anatomy, disease is also get monitored, which is beneficial for effective diagnosis.

Despotovic et al. [2] proposed examination of pictures in medicinal science picture division is a standout among the most basic advances. Picture division is regularly utilized for estimating and envisioning the mind's anatomical structures, for examining cerebrum changes, for depicting obsessive locales, and for careful arranging and picture guided intercessions amid cerebrum MRI.

Bhima and Jagan [3] proposed images in medical science becomes one of the emerging and challenging field because of remarkable growth in image processing. Medical diagnosis is conducted using MRI-based brain medical imaging so that it can exhibit the inner portions of the human brain.

Lal and Aju [4] therapeutic diagnosing framework restorative picture preparing is a standout among the most animating and rising field. Preparing of MRI human cerebrum pictures is one of the fundamental necessities nowadays. The anomaly extricated from the magnetic resonance image (MRI) cut pictures amid determination is depicted. After that irregularity extraction amid, the analysis is performed dependent on the measurement estimations of pictures it very well may be brain tumor, hemorrhage (stroke), edema (brain swelling) and hydrocephalus (water on the cerebrum), and so forth. As per Deshmukh [5], instrument of cerebrum tumor division is only the dissimilarity between various tumor regions from magnetic resonance (MR) pictures. Number of mechanisms are present for the segmentation of brain tumor. As per Selvaraj [6] findings, nowadays, the most threatening disease occurring among man beings is brain tumor. Yet, if brain tumor is diagnosed at the early stages, there are more chances of man beings survival. Also Tambe [7] shows that tumor is an aberrant growth of cells within the brain, which can be the cause of death among people. The main goal of segmentation is to make image easy to understand. This paper provides review on related study of segmentation techniques for segmenting brain tumor from magnetic resonance image.

Manju et al. [8] proposed image segmentation plays vital role in diagnosis and treatment of perilous diseases. The tumor area in brain is properly identified by using different algorithms like seeded region growing and merging K-means, KNN, fuzzy C-means, and related study of all methods presented.

In 2017, the Deshpande and Honade [9] presented an sequential approach tissue to tumor that utilizes MRI images. This approach applies image segmentation with a firefly algorithm. In comparison with SVM linear approach, this shows much better results and less latency.

Emrah et al. [10], Mostafa et al. [11] presented a clustering approach form CT images utilizes the segmentation approach. From MRI images, artificial bee colony approach applied in liver/brain tumor detection and segmentation.

Synthuja et al. [11] proposed swarm intelligence approach for brain tumor recognition. This proposed methodology having good visualization of tumor region. This is further continuous with artificial neural network.

Pathak and Sejwar [11] purposed an technique having two phases, the first stage having histogram analysis approach that allows the growing of initial seed, and the second phase uses cuckoo approach and further threshold optimization. These results are further compared to do the analysis of suggested approach and conformance for betterment.

3 Proposed Methodology

Our approach is composed of two classifier techniques, the first one is bounding box approach in which the frequencies used for detection, and the second approach is Haar used for region of interest detection with feature analysis of inputted image.

3.1 Bisection TDE Mechanism

This approach utilizes the training images with k-dimensions that surrounds the objects and then calculates the V vector using numerous frequencies of the box represented there.

3.2 Flowchart of Purposed Method

The below given flowchart is for our purposed methodology. This is flowchart picturized the different steps involved in realizing the purposed technique.

The approach imitated with inputting an image which we are going to analyze. This image is then converted into frames which are labeled and by applying purposed approach compare with desired area, if matched, then it finished with detection, otherwise the same process applied with next frame.

3.3 Training

This is the important phase in our proposed approach which helps us in feature analysis which further utilizes in region of interest—ROI detection. In training, both bisection as well as Haar approaches are utilized. The ROI is detected on the basis of given preferences mentioned in the approach (Fig. 1).

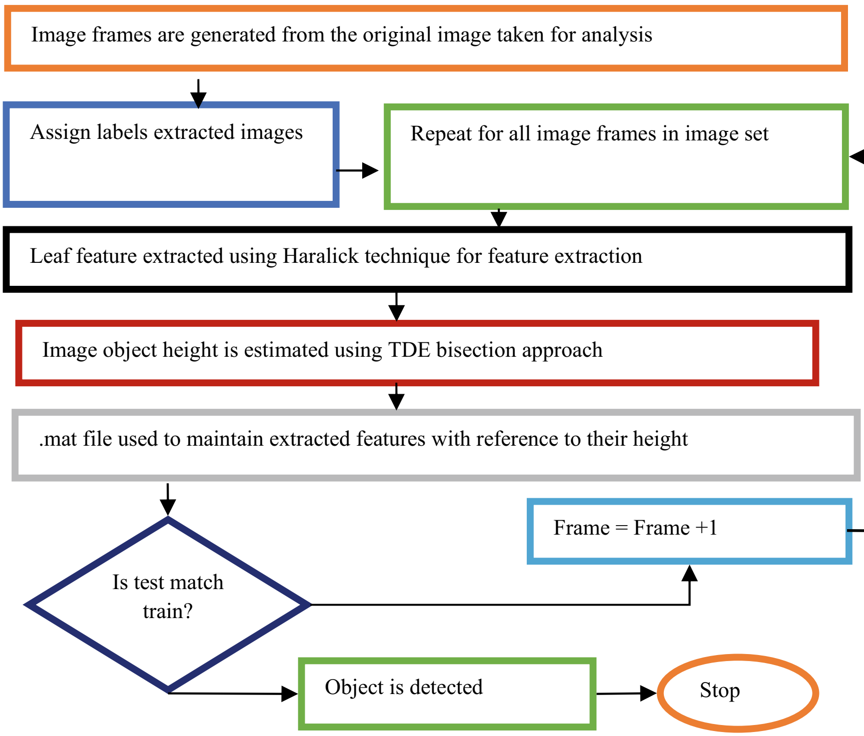


Fig. 1 Proposed approach

3.4 Testing

1. The very initial step is to convert the input image into frames after performing analysis on their proprieties. We applied the standard MATLAB library for performing analysis. The processing format is represented in.avi, and the function available in signal analysis library is mmread(). The same in utilized by us in performing the initial step of proposed approach.
2. After extracting the images, each image is labeled as Image1, Image2, Image3, …, Imagen
3. Loop from first image to last image is applied for processing each image frame.

   a. Mining of the feature of leaf features done through Haralick extraction approach for feature. The result of the same is saved in .mat file format. Based on our assumption that texture feature is in varied represents.
   b. The height of objects from image is estimated using bisection metaheuristic approach.
   c. The .mat file is used here to main the feature extracted corresponding to their height.
   d. Start
      If (Trainingf == Testingf) thenobject foundreturn ();ElseFrame = ++FrameEnd {if statement}End {Looping statement}

      object found
      return ();
      Else
      Frame = ++Frame
      End {if statement}
      End {Looping statement}

   e. Stop

This purposed approach accurately classifies region in various sections. The ROI is also get improved using the TDE, H-bisection classifier. This methodology correctly classifies leaf into distinct categories. This algorithm helps in reduction of non-classified region and much better classification. This approach is a dynamic in nature, then only there is possibilities of data sets analyzes and not only the static images. This approach inherits both supervised as well as un-supervised learning approaches in well form.

4 Experimental Setup and Result

This section is dedicated for concluding the finding of purposed approach. As per our analysis, the use of bisection TDE approach and representations in MSE as well as in PSNR is descripted in itself. The confirmation of purposed approach through

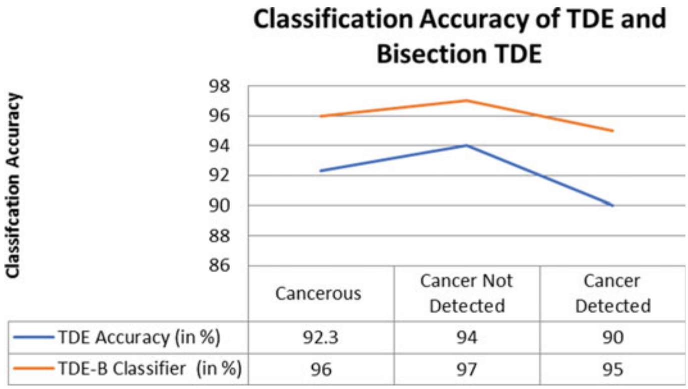


Fig. 2 Classification accuracy of existing and proposed system

classification accuracy is more clarifies our work. The standard utilized in helps in confirmation of correctness of resulted regions from the inputted data sets images. There are four different parameters are used for result representation such as two for positives (false and true) and two for negatives (false and true).

$$\text{Classification}_{\text{Acc}} = (\text{TP} + \text{TN}/\ \text{TP} + \text{TN} + \text{FP} + \text{FN}\ ) \tag{1}$$

True positive is termed as—TP and true negative is represented as—TN result attained throughout experimentation. The accuracy of the resulted classification represented in regions of detection from the inputted data sets. The results represent the frequency of the true positive. These resulted can be changed with time as the consideration of parameters varies with true negative rates. In totality, it can be said that there is not specific symmetry of various categories of classification. This variation in the results of classification is visible there.

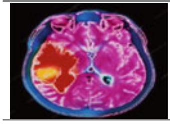
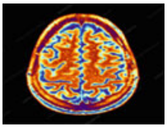
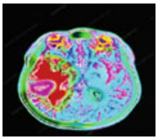
It is depicted in the table above that accuracy in results is 95–97% from that of bisection TDE and from TDE method, and it varies from 90 to 94% (Fig. 2 and Table 1).

There is betterment from 2 to 3% in peak signal to noise ratio as per result of proposed approach. The compared approach having variation from 12 to 13 in their PSNR representation range with respect to our suggested approach the peak signal to noise ratio varies from sixteen to twenty-one only. It is further concluded that the PSNR ratio for the high convergence should be large. The same PSNR is described by the below equation:

$$\text{PSNR} = 10 \log_{10}(\text{MSE}^2/255) \tag{2}$$

Equation 2 shows the PSNR—(peak signal to noise ratio).

Table 1 Classification accuracy

Used image	Classified as	TDE accuracy (%)	TDE-B classifier (%)
 [Kaggle]	Cancerous	92.3	96
	Cancer not detected	94	97
	Cancer detected	90	95

5 Conclusion

The TDE hybrid method with a visual approach used in the proposed system eliminates duplicate features and thus detects cancerous images on a large scale. The result of the proposed system compared to the existing system is better by a limit of 8–10%. The main reason for the improvement was due to the improvement of comparison and measurement of pre-processing and operational performance in addition to classification and classification. As we work in the medical field, the production of rapid results is a need of the hour. This is achieved through the proposed system. The different color features found can be used for real-time images and diagnostics. In future, this method can be tested in comparison with real-time images, and performance can be tested in the GPU for in-depth reading.

References

1. Tirpude N, Welekar R (2013) Automated detection and extraction of brain tumor from MRI images. Int J Comput Appl 77:26–30
2. Despotovic I, Goossens B, Philips W (2015) MRI segmentation of the human brain: challenges, methods, and applications. Comput Math Methods Med 2015:1–23
3. Bhima K, Jagan A (2016) Analysis of MRI based brain tumor identification using segmentation technique. In: 2016 international conference on communication and signal processing (ICCSP). IEEE, pp 2109–2113
4. Anisha ML, Aju D (2014) Abnormality extraction of MRI brain images using region growing segmentation techniques. Int J Enhanc Res Sci Technol Eng 3(8):76–82
5. Deshmukh RD, Jadhav C (2014) Study of different brain tumor MRI image segmentation techniques. Int J Sci Eng Comput Technol 4(4):133

6. Selvaraj D, Dhanasekaran R (2013) Mri brain image segmentation techniques—a review. Indian J Comput Sci Eng (IJCSE) 4(5):0976–5166
7. Tambe P, Thomson J, Surti N Comparative study of segmentation techniques for brain tumor detection. Int J Eng Res Technol (IJERT), vol 5. ISSN (2278-0181)
8. Manju D, Seetha M, Rao KV (2013) Comparison study of segmentation techniques for brain tumor detection. IJCSMC 2(11):261–269
9. Deshpande MS, Pooja S, Honade SJ (2017) Brain tumor segmentation and detection using firefly algorithm. J Electron Commun Eng 12(2)
10. Hancer E, Ozturk C, Karaboga D (2013) Extraction of brain tumors from MRI images with artificial bee colony based segmentation methodology. In: 2013 8th international conference on electrical and electronics engineering (ELECO). IEEE, pp 516–520
11. Mostafa A, Fouad A, Abd Elfattah M, Hassanien AE, Hefny H, Zhu SY, Schaefer G (2015) CT liver segmentation using artificial bee colony optimisation. Procedia Comput Sci 60:1622–1630
12. Preetha MMSJ, Suresh LP, Bosco MJ (2016) Region based image segmentation using cuckoo search algorithm. J Chem Pharm Sci 9(2):884–888
13. Pathak S, Sejwar V (2017) A review on image segmentation using different optimization techniques. Int J Comput Sci Eng 217–221

U-NET Xception: A Two-Stage Segmentation-Classification Model for COVID Detection from Lung CT Scan Images

R. T. Akash Guna, K. Rahul, and O. K. Sikha

Abstract COVID-19 has immensely affected our routine lives and has made us live in a panic driven environment for the past few years. RT-PCR test is the most preferred test for the detection of COVID despite being a time-consuming process. Research has shown that CT scan of lungs can be used to identify COVID infected people in a shorter span of time. The proposed model uses a U-NET-based two staged detection model to segment the abnormal region from the CT scan images. The segmentation module consists of U-Net which produces a binary segmentation mask of regions infected by COVID. The U-Net was trained with dice coefficient loss to improve the structural similarity between ground truth regions and predicted regions. The segmented regions are used to classify COVID contraction. Multiple classification models such as Xception, ResNet, and Inception-ResNet were tested to classify segmented regions. Xception model outperformed other models, produced an accuracy of 98.77%, and thus was chosen as our backbone network. We compare the results obtained on detection using complete CT scans and segmented CT scans to prove the relevance of using segmented regions for COVID prediction.

Keywords CT scan segmentation · COVID detection · U-NET · Xception

1 Introduction

Coronavirus disease (COVID-19) has posed a great threat to mankind. 21.9 crore cases have been reported so far, and the death toll stands at 45.5 lakhs. The real-time

R. T. Akash Guna · K. Rahul · O. K. Sikha (✉)
Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India
e-mail: ok_sikha@cb.amrita.edu

R. T. Akash Guna
e-mail: cb.en.u4cse19302@cb.students.amrita.edu

K. Rahul
e-mail: cb.en.u4cse19328@cb.students.amrita.edu

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_29

polymerase chain reaction (RT-PCR) test has become the standard methodology for COVID-19 detection. RT-PCR test is laborious and requires high-genome detection of viral presence from CT scan lung images is an alternative to the standard testing procedure. Researchers have worked on the detection of COVID-19 from CT scans and have produced numerous classification models for COVID detection. Kavya et al. [1] employed CXR image classification using support vector machine as per the current trends in ML and AI for COVID classification. A modified VGGNet model is used by Anand et al. [2] to classify chest X-ray images. Ahmed T Sahlol's [3] has made use of fractional order marine predators' algorithm for image classification on chest X-ray images in his recent research work. Adnan Saood's [4] experimental results suggest that SegNet-a scene segmentation network can be more accurate for the purpose of classification. The aforementioned classification models reported in the literature suffered from false negatives owing to comparatively small regions of COVID infection in early stages of detection. The large unaffected regions of lungs are capable of misleading the classifier to produce a false negative result. In this paper, we propose to employ a U-NET-based image segmentation to segment COVID affected regions from the CT scan images and then do classification only on the segmented parts. The proposed two-stage segmentation-classification model performed better in terms of classification accuracy.

2 Related Works

Image classification [5] is a process in which an image is classified by comprehending the entire image as a whole. Image classification is widely used in medical imaging. Guangyu Jia's [6] research work has focused on using modified MobileNet for the classification model. Ghousal et al. [7] experimented the use of Bayesian CNN for COVID detection from chest X-ray images. Image segmentation process divides the image into different regions of interest [8–11]. Gerard's [12] experimented on a polymorphic training approach with hierarchical clustering to accurately segment COVID-19 cases. Kai et al. [13] proposed a dual branch combination network for COVID lesion segmentation from lung CT images.

3 Proposed Model

This section describes the proposed model in detail. The model has two modules—a segmentation module and a classification module. The segmentation module consists of a U-Net architecture which takes CT scan images of lungs and segments COVID affected regions to generate a binary segmentation map, where each pixel value specifies whether the corresponding pixel in the input image is affected by COVID. The binary map is then used to segment the COVID affected regions from the input image. The segmented image is then passed into deep learning classification models

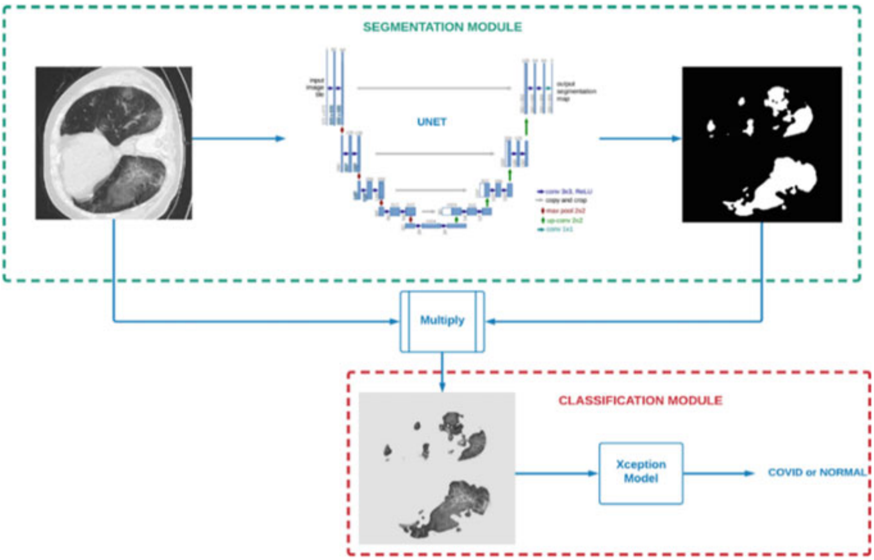


Fig. 1 Proposed architecture. An input image is passed into a U-Net model which segments COVID affected regions using a binary map. The binary map is multiplied with the input image to get the segmented image. The segmented image is passed into an Xception model to perform COVID classification

to classify whether the person is affected by COVID. State-of-the-art classification models such as Xception, ResNet, and Inception-ResNet were explored for the study. Figure 1 visualizes the architecture of the proposed model.

3.1 U-Net

Olaf et al. [14] introduced an enhanced auto-encoder model especifically for biomedical image segmentation called U-Net. It contains an encoder and a decoder. Encoder down samples an input image into a latent space. Decoder performs successive deconvolutions to upsample the latent vector to the size of input image. U-Net possesses skip connections between convolutional and deconvolutional layers which reduces the loss of semantic data caused during the reduction of the input image to a latent space. Figure 2 shows the architecture of U-Net which resembles the English alphabet "U". The proposed model employs dice coefficient loss to train the model. Dice coefficient loss was introduced for image segmentation by Milletari et al. for 3-dimensional medical image segmentation [15].

Dice coefficient loss determines the overlap between two images providing a global loss as shown in Eq. 1.

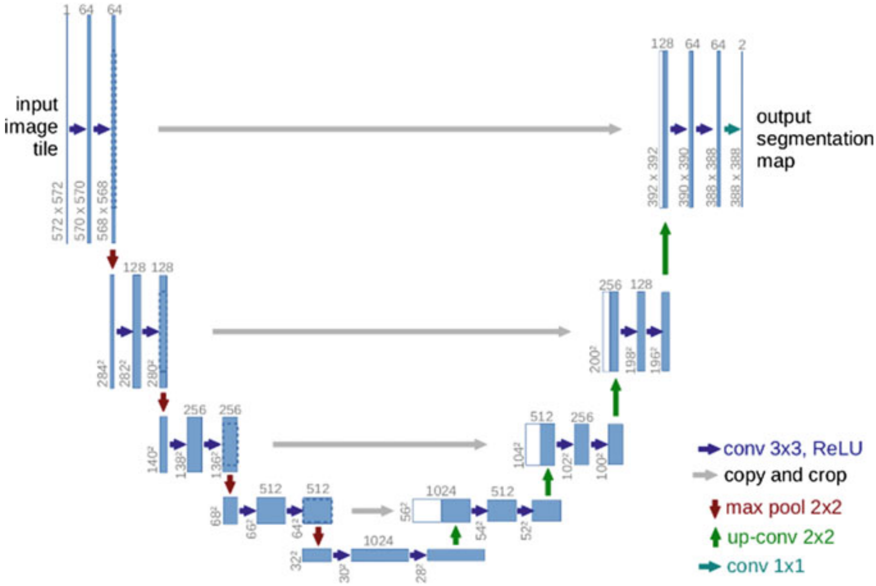


Fig. 2 Architecture of U-Net

$$\text{Dice Loss} = \frac{2 * \sum_i^N p_i * q_i}{\sum_i^N {p_i}^2 + \sum_i^N {q_i}^2} \tag{1}$$

where N is the total number of pixels in the images and p_i and q_i represent a pixel in predicted mask and ground truth mask, respectively.

3.2 Xception

Xception is an extended version of the InceptionV3.The model proposes an interpretation of Inception modules as an intermediate step before the depth wise separable convolution operation: in which the depth and spatial dimensions of a filter can be separated. The feature extraction base of the architecture is formed by 36 convolutional layers, and the layers are structured into 14 modules. Every module except the first and last has linear residual connections around it. Binary cross entropy loss was selected to train the backbone network since COVID identification is a binary classification problem. The loss function is based on how far or close are the probabilities from the actual value. The binary cross entropy loss function is given by Eq. 2.

$$\text{BCE} = -\frac{1}{n} \sum_{i=1}^{n} g_i \, * \, \log p_i + (1 - g_i) \, * \, mathrm{*}log(1 - p_i) \tag{2}$$

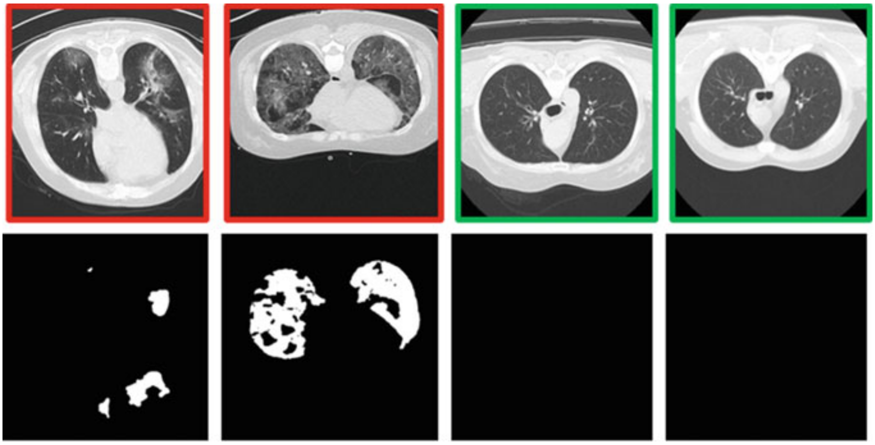


Fig. 3 Sample images from used datasets and their respective ground truth segmentation mask. Green border indicated normal CT scan and red border indicates COVID infected CT scan

Where n is the total number of predictions and p_i and g_i represent predicted class and ground truth class, respectively.

4 Dataset

Two datasets were combined for the analysis. The first dataset [16] consists of CT scans of more than 40 patients. The data were collected and segmented by radiologists from Oslo. The dataset contains 100 CT scans of COVID affected patients. The second dataset contains 829 slices of COVID affected and normal lungs collected from Radiopedia [17]. The dataset has been evaluated by a radiologist who identified that 373 slices were COVID affected slices, and 456 slices were non-COVID lung CT scans. Figure 3 shows some sample CT scans from the database.

5 Results and Analysis

5.1 Segmentation Results

The ground truth map of normal lung CT scans had all pixels filled with one while COVID affected lungs had considerable number of white patches in its binary maps. This trend was replicated by the results from the U-Net model. The predicted masks had a similar structure to that of ground masks. Structural similarity metrices such as PSNR, SSIM, and MSE were used to test U-Net. The scores received on these metrices are shown in Table 1.

The lung CT images of patients with severe COVID infection have bigger regions of abnormality than moderately affected or normal lungs. Moderately, infected lungs showed larger number of disjoint regions of anomaly. Figure 4 illustrates candidate examples of lung CT scan images of people with severe infection, moderate infection, and no infection. From the figure, it is clear that the intensity of infection is clearly distinguishable based on predicted masks and thus improves classification accuracy.

Table 1 Scores received during calculation of image similarity metrics between ground truth mask and predicted mask for the following input CT scans

CT-SCAN					
SSIM	0.972	0.850	0.897	0.938	0.956
PSNR	20.50	11.21	14.13	15.47	17.82
MSE	0.008	0.075	0.038	0.028	0.016

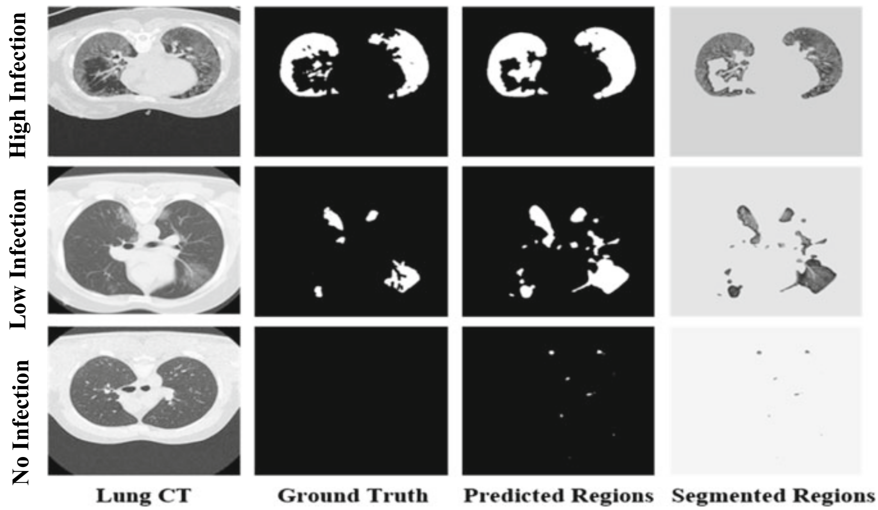


Fig. 4 Tabulation of results received on using U-Net to segment COVID infected regions from lung CT scans

Table 2 Analysis of improvement in COVID detection using infected regions

Model	CT scan	Segmented regions
Xception [18]	0.80	0.9877
Inception-ResNetV2 [19]	0.85	0.9630
ResNet [20]	0.89	0.9600

Table 3 Testing metrices results for classification models

Metrics	Xception	Inception-ResNet	ResNet
Precision	*1.0*	0.88	1.0
Recall	*0.88*	0.77	0.77
F1 score	*0.94*	0.821	0.875

5.2 Relevance of Segmented Results

We trained Inception-ResNet, Xception, and ResNet101 models on CT scan of COVID patients to predict contraction of COVID. We repeated the same process for segmented regions of CT scans, and the results are shown in Table 2. From the table, we could notice that all the models showed a significant improvement on training from segmented results than on complete CT scans. We can also notice that Xception model has outperformed Inception-ResNet and ResNet101 for the classification task**.** Table 3 shows the scores received for classification in terms of precision, recall, and F1 score. The best results are shown as bold. From the table, it is clear that that the proposed two-stage U-NET Xception segmentation-classification framework produces better classification results than other models.

6 Conclusion and Future Works

The paper proposes a two-stage segmentation-classification framework using U-Net and Xception models. The U-Net model was employed to segment the COVID infected regions from lung CT scan images. The segmented regions were further fed into a deep learning model for classification. Using Xception as backbone network, we achieved an accuracy of 98.77%. We have shown that the relationship between intensity of COVID and area of segmented regions helps in improving the prediction results. We then compared the accuracy while using complete CT scans as input to using segmented COVID regions as input. We thus proved the relevance of using segmented regions in aiding the classification model to detect COVID affected patients.

We noticed that area of segmented regions is directly proportional to the intensity of COVID affection. Therefore, we could improve the architecture to produce a fractional number indicating the intensity of COVID affection. Researchers are tied down due to lack of large number of COVID affected samples, thereby limiting the application of current CNN-based COVID detectors. We could test the usability of one short learning architectures as the backbone network to improvise the reliability of the model.

References

1. Garlapati K, Kota N, Mondreti YS, Gutha P, Nair AK (2021) Detection of COVID-19 using X-ray image classification. In: 2021 5th international conference on trends in electronics and informatics (ICOEI), Tirunelveli
2. Anand R, Sowmya V, Vijaykrishnamenon, Gopalakrishnan EA, Soman KP (2021) Modified Vgg deep learning architecture for COVID-19 classification using bio-medical images. In: IOP conference series: materials science and engineering, vol 1084, p 012001
3. Sahlol AT, Yousri D, Ewees AA et al (2020) COVID-19 image classification using deep features and fractional-order marine predators algorithm. Sci Rep 10:15364
4. Saood A, Hatem I (2021) COVID-19 lung CT image segmentation using deep learning methods: U-Net versus SegNet. BMC Med Imaging 21:19
5. Sowmya, Soman KP, Deepika J (2014) Image classification using convolutional neural networks. Int J Sci Eng Res 5(6)
6. Jia G, Lam HK, Xu Y (2021) Classification of COVID-19 chest X-Ray and CT images using a type of dynamic CNN modification method. Comput Biol Med 134:104425. https://doi.org/10.1016/j.compbiomed.2021.104425
7. Ghoshal B, Tucker A (2020) Estimating uncertainty and interpretability in deep learning for coronavirus (COVID-19) detection, arXiv:2003.10769
8. Tyagi P, Singh T, Nayar R, Kumar S (2018) Performance comparison and analysis of medical image segmentation techniques. In: IEEE international conference on current trends in advanced computing (ICCTAC-18), Bangalore
9. Reddy D, Bhavana V, Krishnappa HK (2018) Brain tumor detection using **image** segmentation techniques. In: Proceedings of the 2018 IEEE international conference on communication and signal processing, ICCSP 2018, pp 18–22
10. Sikha OK, Sachin Kumar S, Soman KP (2018) Salient region detection and object segmentation in color images using dynamic mode decomposition. J Comput Sci 25:351–366
11. Sikha OK, Soman KP (2020) Multi-resolution dynamic mode decomposition-based salient region detection in noisy images. SIViP 14(1):167–175
12. Gerard SE, Herrmann J, Xin Y et al (2021) CT image segmentation for inflamed and fibrotic lungs using a multi-resolution convolutional neural network. Sci Rep 11:1455. https://doi.org/10.1038/s41598-020-80936-4
13. Gao K et al (2021) Dual-branch combination network (DCN): towards accurate diagnosis and lesion segmentation of COVID-19 using CT images. Med Image Anal 67:101836
14. Ronneberger O, Fischer P, Brox T (2015) U-net: convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention. Springer, Cham
15. Milletari F, Navab N, Ahmadi S-A (2016) V-net: fully convolutional neural networks for volumetric medical image segmentation. In: 2016 fourth international conference on 3D vision (3DV). IEEE
16. http://medicalsegmentation.com/COVID19/
17. https://radiopaedia.org/articles/COVID-19-3
18. Chollet F (2017) Xception: deep learning with depthwise separable convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition
19. Szegedy C et al (2017) Inception-v4, inception-resnet and the impact of residual connections on learning. In: Thirty-first AAAI conference on artificial intelligence

20. He K et al (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition

AI Integrated Blockchain Technology for Secure Health Care—Consent-Based Secured Federated Transfer Learning for Predicting COVID-19 on Wearable Devices

T. Ravi Shanker Reddy and B. M. Beena

Abstract *COVID-19* has been a major global challenge these days. The pandemic has changed human life, attitude, and behavior. This pandemic added a burden to people's life and health. With the new variants of SARS-CoV-2, a lot of people are even scared of going to the health centers to get the *COVID-19* evaluation in fear of contamination and contagious, which caused the surge in the symptoms at later stages. Data collected across various sources can play an important role in predicting and identifying of *COVID-19* virus based on the models and the classifications of this data using the most sophisticated machine learning models. The concern here is accessing or transferring an individual's data from their personal health devices which defers users' privacy. In the recent past, there are a lot of research that has been done these days on how blockchain can help to securely track and transfer the data across trusted sources. Adding to this, federated learning also is helping on-device data usage without any critical data to be transferred to various external sources. The proposed study directs the stability of frequent health status with the help of wearable devices that capture health metrics like heart rate, blood oxygen levels, breathing rate, muscle activities, stress, emotions, movement patterns, sleep activity, precipitation, and mind/cognitive functions with the introduction of the data streams and models that can seamlessly transfer the data, with the assurance of data integrity, privacy, and control which is the scope of this paper. The usage of both the emerging technologies provides a value addition in terms of health data exchange with effective data distribution with decentralized privacy and computation. We have also introduced a consent-based personal health device registration mechanism on a blockchain consensus network with digital identity to allow and take back controls over who can access their data. We believe that this solution and the implementation would help everyone to predict the possible *COVID-19* infections keeping data privacy at the most priority.

T. Ravi Shanker Reddy (✉) · B. M. Beena
Department of Computer Science and Engineering, Amrita School of Engineering, Bengaluru, India
e-mail: bl.en.r4cse21004@bl.students.amrita.edu

B. M. Beena
e-mail: bm_beena@blr.amrita.edu

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_30

Keywords On-device learning · Secure aggregation · Secure by device consent · Distributed ledger · Training loop

1 Introduction

In today's world, we encounter a novel variety of viruses called coronavirus (*COVID-19*) that brings an acute respiratory infection in the lungs that has caused more deaths across the globe. This pandemic has affected the world massively for more than a year now, where the spread of the virus is so overwhelming for humankind which has not been seen in almost many decades. Now, with the new variants of viruses emerging and given this condition is highly contagious, the ways to detect *COVID-19* among the population remain challenging and also top priority tasks in the health industry[1].

Few primitive methods such as sampling throat and nasopharyngeal swabs are the most feasible way for the diagnosis. For this method, the patient needs to be physically present in the testing centers which gives even more mental agony during the pandemic with the fear of contagiousness. This pandemic, on the other hand, has also brought a lot of inventions into the health and fitness sector by building new applications by integrating artificial intelligence (AI) technology along with traditional health care with the latest infrastructure toward tele-health. Along with machine learning (ML), Io(Health)T devices like wearable devices[3] could alert when there are changes in the user metrics that match *COVID-19*. Prioritizing the safety and well-being of people, data gathered locally across geographies can help the localized bodies and health officials who can put the local area into a containment zone and control and mitigate the spread of the *COVID-19* virus.

The health data of a user should move of their personal devices to the servers that are hosted by the device stakeholders and in turn to various health institutions to scan, evaluate, and determine the possible options[4]. The most common practice for building and training ML models is to curate the data across various real-time stream, and storage sources will raise the following concerns[2].

1. Regulation of data sharing policies while gathering the data from various edge locations.
2. Ensure privacy for the sensitive data.
3. Limited storage infrastructure to collect and avoid data breaches and security issues.
4. Enhanced predictive models, that are trained on a variety of data samples.

A comprehensive study was made which motivated us to propose and implement a solution on how this data has to be securely transferred to the technology organizations, who build applications that help to solve the problem in the need of this hour.

2 Motivation

As we know, our daily activities are more inclined to our human body's health. The industry has moved ahead in developing wearable Io(Health)T automation tools and devices such as smart glasses, wrist bands, and smart bio-devices that also help humans to track and understand their health status[5]. Electronic wearable devices allow physiological and biological signals to be tracked and monitored continuously that can be used in the detection of viruses in the early stages and help to reduce the widespread of both asymptotic and ill cases of *COVID-19*[6].

From the comparative analysis, the following are points that were inferred.

- Wearable devices are so adaptive and convenient so that they need not carry any externally connected devices that are linked like phones. This brings unique remote health opportunities in the prediction and treatment of *COVID-19*.
- Devices are embedded with multiple sensors and algorithms that capture health metrics like heart rate, blood oxygen levels, breathing rate, muscle activities, stress, emotions, movement patterns, sleep activity, precipitation, and mind/cognitive functions. These electronic sensors in the form of tattoos, bio-chips, contact lenses, face masks, and bands as shown in Fig. 1 can help collect physical and biochemical signals [7].

Fig. 1 Wearables types and features

- The convergence of physical and digital devices will enhance the early predictions and identifications of infections and a chance to add value to the behavior and lifestyle patterns of humans.
- Considering the range of smart devices on the human body, there is a huge amount of data that is currently collected and distributed to the central servers across technology and health organizations which may increase the network congestion[8].
- The data sharing process across different organizations sometimes will be impossible due to various privacy regulations, impacting the collection of datasets on the servers. Moreover, getting healthcare services to access these devices is challenging as health care is an exceptionally regulated industry and slow at technology adoption.
- Mostly, the researchers have proposed a blockchain framework for resolving the data privacy over authentication and authorization across the reports. Encryption mechanisms like homomorphic encryption, tokenization, and dual hashing of data too were considered and implemented.
- Blockchain frameworks with confidentiality, auditability, condition-based evaluation using smart contracts, and "Proof of Delivery"-based consensus were being considered and implemented in many healthcare industries.
- Blockchain framework was also considered for inter-party data communications and trust which primarily focused on one–one stakeholder engagement.

Here, considering these studies, an important gap was observed in terms of how humongous data that will be shared, used, and discarded after use for the development of various ML models and applications. There came a thought of

- Why the data should leave the wearable device?
- How do the device stakeholders and healthcare organizations build applications without having the user data?
- How a user is authorized to get the prediction based on his health track status?
- What if the user does not want to be part of this engagement despite owning a smart device?

The following proposed solution details and addresses the problems mentioned above.

3 Proposed Solution

The proposed study considers the data collected by the Io(Health)T-based devices on the daily activities of a common user. This data is the fundamental block for the stakeholders to build AI/ML technologies and utilize, train, and predict the symptoms of *COVID-19* viruses in the body. Unfortunately, there are a few critical challenges in today's wearable health care.

- Distribution of data across devices
- Privacy constraints for data sharing.
- Anonymized, restricted centralized model development.

Following are a few thoughts that arise based on the challenges defined above.

- How to train models that predict *COVID-19* in a distributed way on several wearables without data leaving the devices without data leaving the devices?
- Is it possible to achieve models as strong as those that can be trained on large centralized datasets?
- Can the data transfers adhere to the anti-privacy breach regulations[9]?

We introduce a framework **consent-based secured federated transfer learning for predicting *COVID-19* on wearable devices** with distributed and collaborative model development based on the data collected from multiple wearables to predict *COVID-19* infections.

Consent-based secured federated transfer learning is a process of building and learning a federated learning model based on the decentralized data on the wearable devices that are securely registered with the consent management networks over blockchain.

The framework also supports the comprehension of scope and usage of the data on the wearable to predict the possible *COVID-19* infections in compliance with regulatory standards like HL7(health level 7), PHI(protected health information), CDA(clinical document architecture), and many others. The proposed solution is to bring the trustworthy AI in health care on user wearable devices by keeping the following key notes[10]. The validity, legal bias, and the purpose of the data are tracked.

- Quick, self-regulating suggestions based on the decision making.
- Define data privacy preferences.
- Define the level of security to access the device based on the consents.
- Approval and denial, revoke of consents that were already provided.

This framework uses the blockchain technology in the field of consent and privacy data management.

3.1 Blockchain Technology for Consent Management

Millions of applications and solutions in the healthcare industry at this time were onboarded onto blockchain technologies. Blockchain permits immutable, distributed, and decentralized transactions keeping the core factors of trust and security in design. The transactions are traceable and linked to the history that is computed based on consensus protocols like proof of work, proof of authority, proof of delivery, and many more. The framework integrates cryptography and mathematical solutions like hashing, certificates, public–private key management services, and generation and validations of digital signatures making the blockchain consensus and networks safe and secure for data integrity[11]. The health tracking devices can now be on-boarded onto this technology powered by the stakeholders like device manufacturers and healthcare organizations. Every data record collected can be transformed into a verifiable credential that certifies privacy and security.

These devices bind the customer profile with consent to a set of guidelines and the proof of which is stored in the blockchain through hashes[12]. A controlled self-executing smart contract on this blockchain consensus network enables immeasurable options like automation aspects of consent collection processes, device communications, and data controls. This platform will authenticate the credentials and will be a source to hold the transactions over the consent-provided devices across the geographies[13, 14]. This way, we can securely achieve privacy through secure by device consent. The device consent will be collected as a verifiable credential and will be owned by the device or application owner who can share, deploy, or collect the models that are secure and aggregated. The device owner will have complete control to give or restrict the data for complete privacy.

To set the common rules and guidelines, for consent, the devices have to be first registered and by granting permissions in the controlled environment. Refer Fig. 2. The registered devices together with the various types of data should be accessible to the distributed health services in a secured environment and provide their consent to exchange the models to predict the *COVID-19* infections.

The consent is then validated and stored on a blockchain ledger, later which it is available for health service providers to validate and consult. The only ownership still stays with the user who owns his consent. A consent is first secured and then the application owners can then access the controlled data as recorded and governed by the user. This restricted secure aggregated data is later utilized for AI/ML methodologies and transformations, discussed further.

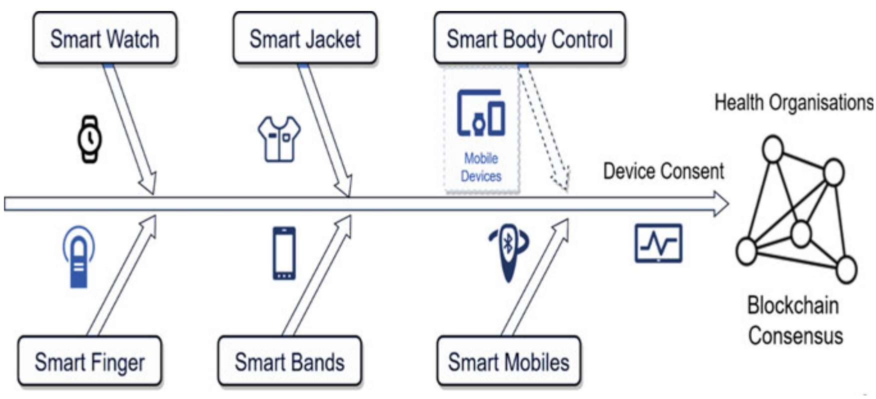


Fig. 2 Users register and give the consent from their wearable over a blockchain network

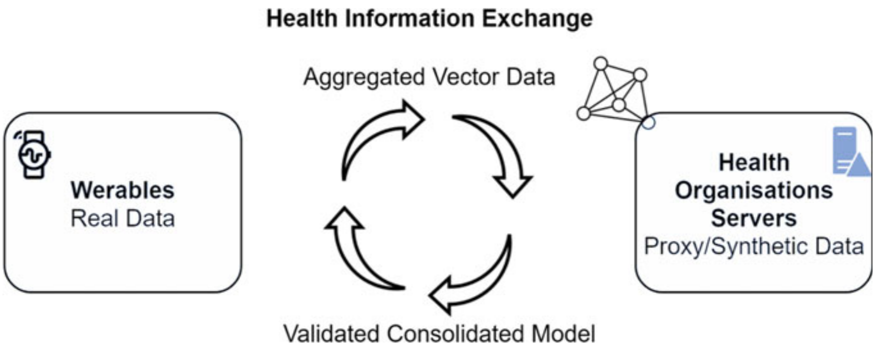


Fig. 3 Secure model aggregation-based health information exchange (HIE) between Io(Health)T and healthcare services

3.2 Machine Learning

Machine learning (ML) is a subset of AI that enables predicting or identifying the outcome based on the existing dataset. This field of AI has advanced in health care to diagnose a wide spectrum of diseases based on the previous patient's diagnosis and treatment history. In traditional ML, the learning and training process is centralized to servers, and the dataset may contain user-sensitive information[15]. Feature extraction networks related to transfer learned convolutional neural network (CNN) are the most common ways to work with limited dataset availability[16].

3.3 Federated Learning

Federated learning is a type of learning process that builds models which is different from distributed machine learning [13]. It works primarily on distributing the computation to various servers. It still needs central servers for training where AI integrated blockchain technology for secure health care the user's data is uploaded, violating the data privacy concerns. In FL, the central servers receiving the information from the edge devices are not the original data, but a trained sub-model. At the same time, the FL also allows asynchronous transmission, and the communication requirements can be appropriately reduced[17].

An iterative process to generate a high-quality model through on-device learning, until it is validated, is called a training loop.

Training loop follows the below flow with secure aggregation of sub-models.

- These wearable devices register on to a blockchain network with a trusted service provider providing the consent that is self-controlled by the user to download the shared prediction model that is trained on a proxy data.

- The devices improve the downloaded base model by learning from the personal data available on the devices.
- Federated Compute runs locally with a focused collection and compute the vectors of the device reports.
- Devices then push only the bias and weights along with the aggregates which are also known as federated means but not exactly the data complying to the minimalistic disclose of data as needed[18].
- Devices aggregate the changes as a small incremental model and send to the servers.
- Understands the learning rates, hyperparameters, noise reduction, performance monitoring and is averaged instantly with the other increments collected from other users as well and that improves the shared prediction model. See Fig. 3.
- Servers on remote regenerate the synthetic data based on the discrete models then combine for validation and feedback.
- Finally, servers prepare the upgraded release of the tested, incremented models for the approved devices to download and predict the health situations.

The flow repeats over time. See also Fig. 4. This way, all the training data stays on the device, and no individual datasets are stored in the servers which maintains privacy with the advantage of early detection of the infections[19].

3.3.1 Advantages

- Data privacy—The data is localized to the user's device, and the consent is provided on the device for local computation[20].
- Low throughput—The models are trained and predicted locally based on the individual device's data.
- Accuracy and validity—The model is trained on local data on top of the proxy data, so the results come out to be more accurate and personalized. The local data is cleaned on a regular basis.
- Power optimization—Model training locally on the devices will help in power optimization.
- Minimal hardware compatibility—Limited hardware configurations are sufficient to run the model building and learning on wearable devices.

3.3.2 Challenges

- Large-scale distribution—Collection of data is across geographies, culture which makes connectivity and model distribution complex.
- Limited communication—Federated blockchain networks consist of multiple types and number of devices which has an impact on the communication to the central servers for fetching improved models.

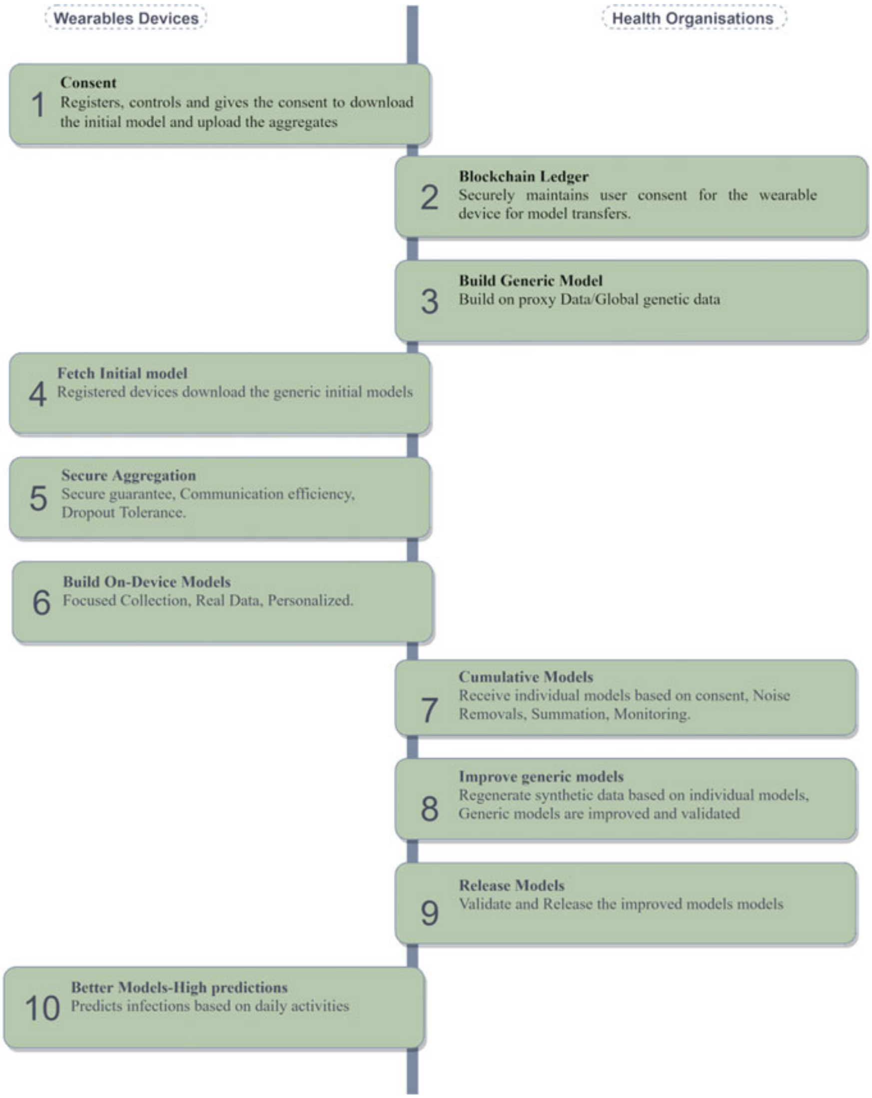


Fig. 4 Sequence flow of **training loop**

- Data diversity and quality—Size and quality of data collected on various devices may add bias to the model training.
- Ephemeral data—Datasets do not belong on the devices for re-validations.
- Model updates—Regular updates are necessary for keeping the models up to date to predict the viruses. In case of any device or connectivity failures, the data/models will be stale for further enhancing of predictions.

4 Conclusion

Considering the rapid spread of *COVID-19* viruses with new mutant variants like Delta, Omicron coming up, it is highly impractical for individuals to get a regular diagnosis. The proposed solution defines the stability of frequent health status with the help of wearable devices that capture health metrics like heart rate, blood oxygen levels, breathing rate, muscle activities, stress, emotions, movement patterns, sleep activity, precipitation, and mind/cognitive functions. The study also implies having a safe and secure prediction or detection of *COVID-19* viruses in humans and driving the state of health.

It is understandable that initially, it is a very difficult task to identify and compare *COVID-19* from various other viral infections. So a regular retrained model will help to improve the prediction as and when a study to understand the variants is developed. These predictions methods would require huge datasets to train the models for every new strain of viruses. Transferring personal data to external health providers is considered a privacy breach, and the solution is to have federated on-device learning based on secure aggregation. Each individual can control what gets controlled by a blockchain-based consent management and verifiable credentials that enables strict controls on who can access the data, artifacts like aggregated vector data, and individually trained models.

In case if there are any true positive predictions, the individual can take supervised medications. The local governing bodies and health officials can put the areas where there are a lot of positive cases into a containment zone to control and mitigate the spread of the *COVID-19* virus.

5 Future Enhancements

Considering the usage of these highly expensive wearable devices, this solution can only be restricted to the affluent segment of the world. Most of the developing countries are still struggling for bread and butter for whom these solutions may not be at reach. Access to economical wearable devices should be considered by the health care and technology organizations to promote new ideas which can help access and train on huge datasets securely. Identifying the patterns and the design solutions can help predict every new strain of viruses and suggest the ideas and possible treatment plans based on the behavioral patterns of the users with wearable devices.

Identity, auditability, and consent management for individuals across multiple devices at a scale are also the scope for future studies.

References

1. Garlapati K, Kota N, Mondreti YS, Gutha P, Nair AK (2021, June) Detection of COVID-19 Using X-ray image classification. In: 2021 5th international conference on trends in electronics and informatics (ICOEI). IEEE, pp 745–750
2. Tech in 3, science and Philosophy; what is federated learning? Science and philosophy, medium, 20 Nov 2020
3. Qayyum A, Ahmad K, Ahsan MA, Al-Fuqaha A, Qadir J (2021) Collaborative federated learning for healthcare: multi-modal covid-19 diagnosis at the edge. arXiv:2101.07511
4. Kakarlapudi PV, Mahmoud QH (2021) Feb) A systematic review of blockchain for consent management. Healthcare 9(2):137
5. Ates HC, Yetisen AK, Güder F, Dincer C (2021) Wearable devices for the detection of COVID-19. Nat Electron 4(1):13–14
6. Seshadri Dhruv R, Davies Evan V, Harlow Ethan R, Hsu Jeffrey J, Knighton Shanina C, Walker Timothy A, Voos James E, Drummond Colin K (2020) Wearable sensors for COVID-19: a call to action to harness our digital infrastructure for remote patient monitoring and virtual assessments. Front Digital Health 2:8
7. Rodrigues JJ, Segundo DBDR, Junqueira HA, Sabino MH, Prince RM, Al-Muhtadi J, De Albuquerque VHC (2018) Enabling technologies for the internet of health things. IEEE Access 6:13129–13141
8. Renugadevi N, Saravanan S, Naga Sudha CM, Tripathi P (2021) IoT-enabled applications and other techniques to combat COVID-19. In: Kautish S, Peng SL, Obaid AJ (eds) Computational intelligence techniques for combating COVID-19. EAI/Springer Innovations in Communication and Computing. Springer, Cham. https://doi.org/10.1007/978-3-030-68936-0_7
9. Kasyap H, Tripathy S (2021) Privacy-preserving decentralized learning framework for healthcare system. ACM Trans Multimedia Comput Commun Appl 17(2):24. DOIurl-https://doi.org/10.1145/3426474
10. Kumar R, Khan AA, Kumar J, Zakria A, Golilarz NA, Zhang S, Zheng C, Wang W (2021) Blockchain-federated-learning and deep learning models for covid-19 detection using CT imaging. IEEE Sens J
11. Mamo N, Martin GM, Desira M, Ellul B, Ebejer JP (2020) Dwarna: a blockchain solution for dynamic consent in biobanking. Euro J Human Genetics 28(5):609–626
12. Sivasangari A, Sonti VJKK, Poonguzhali S, Deepa D, Anandhi T (2022) Security framework for enhancing security and privacy in healthcare data using blockchain technology. In: Khanna A, Gupta D, Bhattacharyya S, Hassanien AE, Anand S, Jaiswal A (eds) International conference on innovative computing and communications. Advances in intelligent systems and computing, vol 1387. Springer, Singapore. https://doi.org/10.1007/978-981-16-2594-7_12
13. Rantos K, Drosatos G, Kritsas A, Ilioudis C, Papanikolaou A, Filippidis AP (2019) A blockchain-based platform for consent management of personal data processing in the IoT ecosystem. Secur Commun Netw
14. Nair A, Kapoor C, NB SS, Balamurugan S (2021, June) Exploring blockchain enabled smart community with electric vehicles. In: 2021 10th IEEE international conference on communication systems and network technologies (CSNT). IEEE, pp 745–750
15. Hu K, Li Y, Xia M, Wu J, Lu M, Zhang S, Weng L (2021) Federated learning: a distributed shared machine learning method. Complexity
16. Chen Y, Qin X, Wang J, Yu C, Gao W (2020) Fedhealth: a federated transfer learning framework for wearable healthcare. IEEE Intell Syst 35(4):83–93
17. Harini N, Ramji B, Sowmya V, Krishna Menon V, Gopalakrishnan EA, Sajith Variyar VV, Soman KP (2021) Multi-task data driven modelling based on transfer learned features in deep learning for biomedical application. Innovations in computer science and engineering. Springer, Singapore, pp 185–193
18. Kim H, Park J, Bennis M, Kim SL (2019) Blockchained on-device federated learning. IEEE Commun Lett 24(6):1279–1283

19. Ramesh TK, Shashikanth A (2020) A machine learning based ensemble approach for predictive analysis of healthcare data. In: 2020 2nd Ph.D. colloquium on ehically driven innovation and technology for society
20. Rajesh G, Raajini XM, Dang H (eds) (2021) Industry 4.0 interoperability, analytics, security, and case studies. CRC Press
21. Rajmohan A, Ravi A, Aakash KO, Adarsh K, Raj AD, Anjali T (2021, March) CoV2eX: a COVID-19 website with region-wise sentiment classification using the top trending social media keywords. In: 2021 sixth international conference on wireless communications, signal processing and networking (WiSPNET). IEEE, pp 113–117
22. Arul R, Al-Otaibi YD, Alnumay WS, Tariq U, Shoaib U, Piran MJ (2021) Multi-modal secure healthcare data dissemination framework using blockchain in IoMT. Personal Ubiquitous Comput:1–13

Deep Feature-Based COVID Detection from CT Scan Images Using Support Vector Machine

S. Lokesh Sai Phani Babu, U. Sri Ranganath, P. Bharath Anuj, C. Divyanth, and O. K. Sikha

Abstract Coronavirus (COVID-19) is an air-borne disease that has affected the lifestyle of people all around the world. Tracing patients infected with coronavirus has become a difficult process because of the limitation of tests based on reverse transcription-polymerase chain reaction (RT-PCR). Recently, methodologies based on imaging have been proposed by various researchers especially using deep learning-based models for the detection of COVID infection. This paper analyzes the effectiveness of deep features for COVID detection from CT scan images. Deep features were extracted from the final layers of deep learning models which are then fed into machine learning frameworks for classification. Transfer learned features obtained from ResNet50, Inception V3, and EfficientNetB7 were employed for the study. A combination of Inception V3 and SVM gave the best accuracy of 86.12 and precision and recall with 83.11 and 80.44, respectively. These results are comparable to recent transfer learning approaches and architecture that is about to be discussed is having an advantage of minimized time when compared to traditional deep learning approaches.

1 Introduction

COVID-19 (coronavirus disease) is a highly contagious disease which is caused by SARS-CoV-2 (severe acute respiratory syndrome coronavirus) [1]. This virus is highly communicable and can be transferred to other people by direct contact or indirect contact. According to WHO statistics, currently, 226,844,344 people have been affected by COVID-19 and there were 4,666,334 deaths across the globe [1]. In 2021, many new contagious and deadly variants of SARS-CoV-19 like Alpha (B.1.1.7) and Delta (AY.1) have been reported in countries like Brazil, UK, and India [2]. These new variants have increased mortality and infection rates in many

S. Lokesh Sai Phani Babu · U. Sri Ranganath · P. Bharath Anuj · C. Divyanth · O. K. Sikha (✉)
Department of Computer Science and Engineering, Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, India
e-mail: ok_sikha@cb.amrita.edu

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_31

countries when compared to the previous variants. According to WHO, the most common symptoms are fever, cough, loss of taste, and smell [3].

One of the causes for the rapid development of COVID in densely populated nations such as India is a lack of efficient tools capable of reliably diagnosing COVID in a short period of time. Out of many COVID diagnosing techniques, scientists have declared RT-PCR (reverse transcription-polymerase chain reaction) as the standard procedure for diagnosing COVID-19 as this is one of the most successful identification tests till date [4]. The RTPCR test demands for a huge number of certified personnel, large equipped laboratories and most importantly it requires a minimum of 5 hours of time [5]. Considering the drawbacks of this diagnosis method, scientists started to work on different techniques to detect COVID-19 infection. Apart from laboratory methods such as antigen deduction, antibody detection, and point-of-care test, there are studies that explore various imaging techniques such as CT (computed tomography) scan images and X-ray images.

CT scans have been used extensively in the diagnosis of COVID-19 infections. When compared to other major diagnostic procedures such as RT-PCR, it has a high sensitivity rate. In rare cases, persons have been diagnosed with COVID-19 disease, yet CT imaging shows that they are not infected [4, 6]. The primary disadvantage of CT scan-based diagnosis is that it requires skilled radiologists to work and takes time. As a result, we cannot rely on this in an emergency. This underlines the requirement for an AI model capable of automating and detecting COVID in a short amount of time while providing superior performance. In addition to accurate COVID identification, the model's time complexity is critical. In comparison to existing deep convolutional neural networks, the proposed model is less complex in terms of learnable parameters and provides faster and accurate prediction.

The proposed model uses advanced deep learning and machine learning models for the accurate detection of COVID from CT scan images [7], which have been proved to be effective in the field of medical imaging. Various transfer learned models ResNet50V2, Inception V3, and EfficientNetB7 are explored to extract features from CT scan images which are then fed into using Machine learning models which are faster than many deep learning models for final classification. The prime contributions of this paper are summarized as follows:

- Deep features were extracted from the deep convolutional models such as Inception V3, ResNet50V2, and EfficientNetB7.
- Supervised machine learning models: SVM, Decision tree, and Random Forest were learned on the extracted deep features for COVID-19 detection.
- Effect of preprocessing on deep feature-based classification is analyzed.

2 Related Works

Medical image analysis flourished buoyed by the advancement of deep learning models [8], such as Residual Network (ResNet) and DenseNet [9]. Yang et al. [8]

proposed two deep learning models to assist us in the process of detecting COVID-19, a DenseNet-based model, and a ResNet-based model. Yang et al. [8] created a dataset by collecting 760 preprints of COVID-19 patients from medRxiv and bioRxiv. With the help of MuPDF, they have extracted minute information and preserved the quality of preprints. Finally, a collection of 349 CT images labeled as positive training samples is generated, and 397 non-COVID-CT scan samples were also added to make a complete classification dataset. The DenseNet-169-based model was then trained on the dataset which attained an accuracy of 69.8, 79.5, and 57.8% on the COVID-seg, COVID-CT-349, and COVID-CT-118 Datasets [8], respectively. Similarly, ResNet50 has attained 66.3, 77.4, and 60.4% on the three datasets as mentioned above. The authors created a lung mask and lesion mask for each CT image and combined COVID-seg and COVID-CT-349 Datasets to attain an accuracy of 87.1%. In [10], the authors proposed a light CNN without bypass configuration for the detection of COVID-19. Instead of bypass configuration, they have added a transpose convolutional layer to the final Custom Fire Module. This design has proved to achieve better accuracy compared to the SqueezeNet model. Silva et al. [11] used EfficientNet for detecting COVID-19 diagnosis in CT images with a voting-based approach and cross-data analysis. In this approach, the images from a particular patient are classified into groups based on voting. Horry et al. [12] have taken three types of images for detecting COVID-19, i.e., X-rays, Computerized Tomography (CT images), and Ultrasound Images. The author has tested the dataset on the optimized VGG-16 model. The experimental results indicated that ultrasound images provide better accuracy than X-rays and Computerized Tomography images. The model has attained a precision of 86%, 100%, and 84%, respectively, on X-rays, ultrasound, and CT images. Hussian et al. in [13] has proposed a 22-Layer CNN architecture called Coro-Det. The proposed model is developed to evaluate accurate diagnostics for 2 class Classification (COVID-19, Normal), 3 class Classification (COVID-19, Normal, and Non-COVID Pneumonia), and 4 class Classification (COVID-19, Normal, Non-COVID viral pneumonia, and Non-COVID Bacterial pneumonia). The model has achieved an accuracy of 99.1%, 94.2%, and 91.2% on 2 class classification, 3 class classification, and 4 class classification, respectively. Diaz-Escobar et al. in [14] used the publicly available dataset POCUS encompassing 3360 ultrasound images of Normal, COVID-19, and pneumonia patients. The author has used pre-trained models including ResNet50, Inception V3, VGG-19, and Xception and evaluated the dataset considering three classes (COVID-19, Normal, and Pneumonia) and two classes (COVID-19 and Normal). Inception V3 has outperformed remaining state-of-the-art models by achieving the best average accuracy of 89.1, best balanced accuracy of 89.3, and area under the receiver operating curve (AUC) of 97.1%. In our work, we have used the COVID-19 CT Dataset that comprises 746 CT Scans of COVID and Non-COVID. We have extracted deep learning features from pre-trained convolutional neural networks like ResNet50V2, Inception V3, and EfficientNetB7. We have used machine learning models to evaluate the features extracted from the deep learning models. The primary aim of this work is to measure the efficiency of Machine Learning models on Deep Learning features.

Table 1 Details of COVID-19 CT dataset [8]

Category	COVID-19 data	Non-COVID-19 data	Total data
Train set	253	292	545
Test set	96	105	201

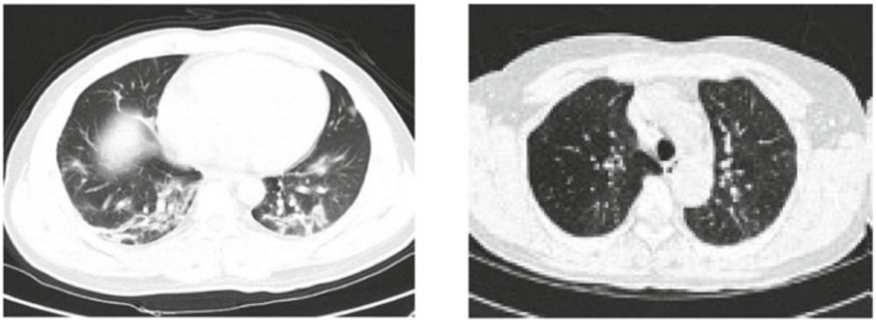


Fig. 1 The left image represents a CT scan of a person affected with COVID-19, and the right image represents a CT scan not affected with COVID

3 Methodology

3.1 Dataset

COVID-19 CT Dataset proposed by Yang et al. [8] is employed for the proposed study. The dataset comprises 746 CT images of COVID-19 patients (from 216 patients) from medRxiv and bioRxiv. To preserve the quality of images, the authors have used PyMuPDF to extract Low-Level structure information of the pdf files of preprints and locate all the embedded figures. The final dataset includes 349 CT images labeled as positive samples and 397 CT images as negative samples, i.e., Non-COVID-CT images. The maximum, average, and minimum height of the images are 1853, 491, and 153 whereas the minimum, average, and maximum width of the images are 124, 383, and 1485. The details of the dataset is tabulated in Table 1, and the sample images are shown in Fig. 1.

3.2 Data Augmentation and Preprocessing

As mentioned in the previous section, the dataset consists CT scan images of different sizes. All the images were reshaped to (224,224) to make it uniform. Various data augmentation techniques such as Random Vertical Flips, Horizontal Flips, Rotation

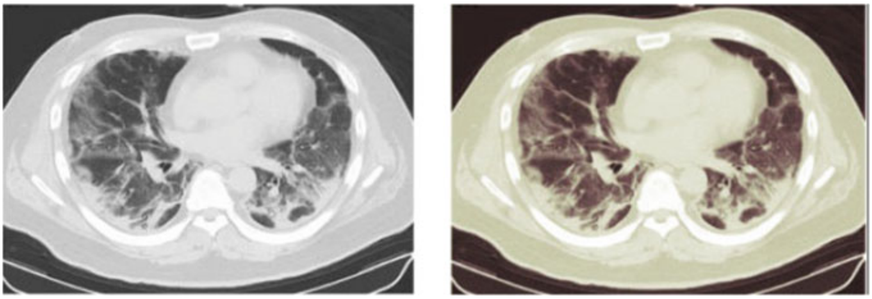


Fig. 2 The left image shows the original image whereas the right image is after applying pseudo coloring and dimension normalization to the original image

of images from 0 to 10 degrees, height shift range, width shift range, and Normalization of images between 0.0 and 1.0 degrees were applied on the dataset to avoid overfitting. To further enhance the quality of the images, pseudo coloring [15, 16] is employed, which gave a better classification accuracy also. Figure 2 shows the original image and the pseudo-colored image.

4 Proposed Architecture

A Convolutional Neural Network (Conv Net) is a special category of neural network architectures that takes an image as input, assigns importance to distinct aspects in the image, and can distinguish between them [5, 9]. They have made significant progress toward improved image classification performance. The proposed model exploited various state-of-the-art CNN models for classification such as Inception V3, ResNet50V2, and EfficientNetB7 for extracting features. The high-dimensional nature of the transfer learned deep features produced from multiple deep convolutional models could have a negative impact on the classification model. This unveils the limitation of this study. High-level features obtained from the final FC layer of the aforementioned deep learning models are then fed into machine learning models to analyze how well the ML models are performing on the deep learning features. The workflow of the proposed model is shown in Fig. 3.

5 Results and Discussions

This section compares the effectiveness of transfer learned features from state-of-the-art models such as EfficientNetB7, Inception V3, and ResNet50V2 for classification. The extracted feature vectors were fed into advanced machine learning models like Support Vector Machine (SVM), Decision Tree, and Random Forest for prediction.

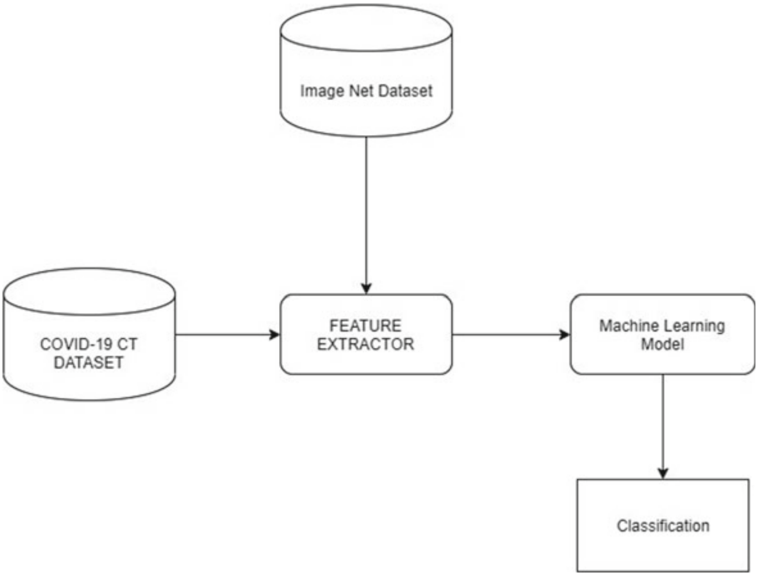


Fig. 3 Workflow of the proposed architecture

The experiment was carried out on the Google Colab Pro which provides NVidia Tesla T4 or P100 GPU. Colab Pro is chosen for experimentation as it provides a longer GPU when compared to Colab and has a high computational power that is required for the experiments. As the dataset is balanced, we have used evaluation metrics such as accuracy (Eq. 1), precision (Eq. 2), and recall (Eq. 3) that are calculated using the values extracted from the confusion matrix for the experimental analysis.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{1}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

We got the best results using a combination of Inception V3 and SVM out of all the Transfer Learning Models we tried. Tables 2 and 3 show the results obtained with and without image preprocessing as explained in Sect. 3.2. On the SVM model, we achieved an accuracy of 86.12%, 85.24%, and 78.19% with the Inception V3, ResNet50V2, and EfficientNetB7, respectively. According to Tables 2 and 3, there is an increase in accuracy for all three models after applying to preprocess.

The proposed model was compared against the state-of-the-art classification models for the COVID classification task. The obtained results are tabulated in Table

Table 2 Performance comparison of various combinations of deep learning and machine learning models on classification without Image preprocessing

	Inception V3			ResNet50V2			EfficientNetB7		
	Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall
SVM	85.24	87.34	90.70	83.80	88.14	85.65	77.81	83.90	79.19
Decision tree	75.12	79.002	76.76	74.96	81.50	76.56	68.70	74.06	76.16
Random forest	78.56	85.49	78.58	79.30	85.17	79.6	78.31	78.60	88.48

Table 3 Performance comparison of various combinations of deep learning and machine learning models on classification with Image preprocessing

	Inception V3			ResNet50V2			EfficientNetB7		
	Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall
SVM	86.12	87.92	89.96	85.24	85.25	87.27	78.19	83.43	81.14
Decision tree	76.70	82.62	82.62	75.34	75.55	75.55	73.72	77.57	77.57
Random forest	80.42	81.61	78.93	78.93	78.98	78.98	72.24	87. 87	87. 87

Table 4 Performance comparison of the proposed architecture with the existing approaches

	Accuracy	Precision	Recall
Jangam et al. [17]	84.73	0.791	0.928
Polsinelli et al. [10]	83.00	0.817	0.850
Silva et al. [11]	**87.68**	0.939	0.795
Proposed architecture	**86.12**	83.11	80.44

4. The proposed model produces equivalent results (in terms of accuracy) to the benchmark models in the literature, as shown in Table 4.

6 Conclusion

There are possibilities that patients affected with COVID might end up damaging their lungs permanently, which can lead to death after a long time. In this paper, we employ image features extracted from various deep learning models for classification. The extracted features were fed into machine learning models for classification, as machine learning models are faster than deep learning models but lack the ability to extract features. With the COVID-19 dataset, the combination of Inception V3 and SVM performed the best, with an accuracy of 86.12%.

References

1. Cucinotta D, Vanelli M (2020) WHO declares COVID-19 a pandemic. Acta Biomed 91(1):157–160. https://doi.org/10.23750/abm.v91i1.9397
2. Janik E, Niemcewicz M, Podogrocki M, Majsterek I, Bijak M (2021) The emerging concern and interest SARS-CoV-2 variants. Pathogens 10:633. https://doi.org/10.3390/pathogens10060633
3. Lai CKC, Lam W (2021) Laboratory testing for the diagnosis of COVID-19. Biochem Biophys Res Commun 538:226–230. ISSN 0006-291X. https://doi.org/10.1016/j.bbrc.2020.10.069
4. Tahamtan A, Ardebili A (2020) Real-time RT-PCR in COVID-19 detection: issues affecting the results. Expert Rev Mol Diagnost 20(5):453–454.https://doi.org/10.1080/14737159.2020.1757437
5. Tamuly S, Jyotsna C, Amudha J (2019) Deep learning model for image classification. In: 3rd International conference on computational vision and bio inspired computing, (ICCVBIC 2019), RVS Technical Campus, Coimbatore
6. Subbiah U, Kumar RV, Panicker SA, Bhalaje RA, Padmavathi S (2020) An enhanced deep learning architecture for the classification of cancerous Lymph node images. In: 2020 Second international conference on inventive research in computing applications (ICIRCA), Coimbatore, India, 2020, pp 381–386. https://doi.org/10.1109/ICIRCA48905.2020.9183250
7. Ramachandran R, Rajeev DC, Krishnan SG, Subathra P (2015) Deep learning—an overview. Int J Appl Eng Res 10:25433–25448
8. Yang X et al (2020) COVID-CT-dataset: a CT scan dataset about COVID-19
9. Vinayakumar R, Soman KP, Poornachandran P (2017) Applying convolutional neural network for network intrusion detection. In: 2017 International conference on advances in computing, communications and informatics (ICACCI). IEEE, New York, pp 1222–1228
10. Polsinelli M, Cinque L, Placidi G (2020) A light CNN for detecting COVID-19 from CT scans of the chest. Pattern Recogn Lett 140:95–100
11. Silva P et al (2020) COVID-19 detection in CT images with deep learning: a voting-based scheme and cross-datasets analysis. Inf Med Unlocked 20:100427
12. Horry MJ et al (2020) COVID-19 detection through transfer learning using multimodal imaging data. IEEE Access 8:149808–149824. https://doi.org/10.1109/ACCESS.2020.3016780
13. Hussain E, Hasan M, Rahman MA, Lee I, Tamanna T, Parvez MZ (2021) CoroDet: a deep learning based classification for COVID-19 detection using chest X-ray images. Chaos, Solitons Fractals 142:110495, ISSN 0960-0779. https://doi.org/10.1016/j.chaos.2020.110495
14. Diaz-Escobar J, Ordóñez-Guillén NE, Villarreal-Reyes S, Galaviz-Mosqueda A, Kober V, Rivera-Rodriguez R et al (2021) Deep-learning based detection of COVID-19 using lung ultrasound imagery. PLoS ONE 16(8):e0255886. https://doi.org/10.1371/journal.pone.0255886
15. Liang S, Liu H, Gu Y et al (2021) Fast automated detection of COVID-19 from medical images using convolutional neural networks. Commun Biol 4:35. https://doi.org/10.1038/s42003-020-01535-7
16. Sachin Saj TK et al (2019) Facial emotion recognition using shallow CNN. In: Symposium on machine learning and metaheuristics algorithms, and applications. Springer, Singapore
17. Jangam E, Barreto AAD, Annavarapu CSR (2021) Automatic detection of COVID-19 from chest CT scan and chest X-Rays images using deep learning, transfer learning and stacking. Appl Intell. https://doi.org/10.1007/s10489-021-02393-4

Ship Detection from Satellite Imagery Using Deep Learning Techniques to Control Deep Sea Oil Spills

Mohamed Fuad Amin Mohamed Jamal, Shaima Shawqi Almeer, and Sini Raj Pulari

Abstract Our planet Earth is presently being disturbed by a variety of environmental concerns. One of the top critical environmental issues affecting our planet's ecosystem is oil spills. Oil spills mostly occur due to ship leakage which highly influences our food supply chain and leads to a high-level drop in the economic division. Therefore, monitoring and tracking those vessels are extremely vital to determine the responsible ships for the occurrence of an incident. This study revolves around an implementation of an automated ship detection software application by building a high-level algorithm that embeds deep learning networks. The algorithm is built in a way that can predict and classify vessels from high-resolution satellite images with 98.5% accuracy.

Keywords CNN · Oil spills · Sea pollution · Deep learning · Satellite imagery

1 Introduction

The Arabian Gulf is known to be a cross-national area within the Middle East. Apart from being a region of tremendous strategic and economic importance, the Arabian Gulf is a common territory of local countries. Bahrain, Saudi Arabia, Oman, Qatar, the United Arab Emirates, and Kuwait are all part of the Arabian Gulf region.

Ever since the discovery of crude oil and associated export operations, the Arabian Gulf has suffered immensely from hazardous oil-related activities such as excavation, oil filling, ballast barrel dumping, exploration, mining, and manufacturing. The is

M. F. A. M. Jamal · S. R. Pulari (✉)
Department of ICT, EDICT, Bahrain Polytechnic, Isa Town, Kingdom of Bahrain
e-mail: sini.raj@polytechnnic.bh

M. F. A. M. Jamal
e-mail: 201501325@student.polytechnic.bh

S. S. Almeer
Space Data Analysis Laboratory, National Space Science Agency, Al Hidd, Kingdom of Bahrain
e-mail: shaima.almeer@nssa.gov.bh

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_32

because, all the countries within the Arabian Gulf have large crude oil deposits and are well-known petroleum product exporters.

Therefore, the Arabian Gulf is recognized as a major hub for the worldwide oil industry. Consequently, oil exploration, production, and transportation pose a constant threat to our marine environment, resulting in a variety of harmful implications such as oil spills. As a result, human society, maritime environments, and other mammals have suffered a variety of losses. Furthermore, this would lead to a significant reduction in fish productivity, tourism, and economy. However, until the second part of the twentieth century, people were not aware of the ramifications or negative effects that environmental contamination could have.

Oil spills in the Arabian Gulf are generated by a variety of factors, the most of which are connected to unintentional accidents involving oil tankers, offshore platforms, offshore drilling, spills, and leaks from pipelines and ships [1].

Tanker oil spills have been the most frequent source of marine pollution. Oil ships have been accused of unlawfully discharging pollution into the oceans on several circumstances. Oil spills can spread across large areas, causing destruction of the natural habitats of species like coral reefs and algae [2]. In addition, those spills have also been linked to the development of diseases and the introduction of hazardous compounds that can alter plant and animal genetic makeup. Oil pollutants, in most situations, threaten marine species' capacity to seek food. Moreover, the hazardous polycyclic aromatic hydrocarbons found in crude oil can be hard to extract and could persist for many years [3].

As a response, considering ships have such a significant impact on our maritime ecosystem, it is critical to be able to track and detect the movement of oil ships. Furthermore, with the advancement of remote sensing imaging and artificial intelligence technology, it is currently easy to visualize and monitor the wide-range sea surface, by detecting the targeted object, resulting in a surge in research interest in marine level detection and classification technology, considering it one of the most crucial, cost-effective, and effective techniques of monitoring and analyzing the environment.

2 Related Works

Ship detection has been a live research topic in the field of Satellite remote sensing recently. As mentioned before, oil spills and its related threats affects have an adverse effect to the marine life population. Identifying objects from satellite imagery is a tedious task specifically when it comes to different types of satellites and its various resolutions to deal with; it is quite a difficult task to identify objects with expected accuracy. A log of wood floating in the sea and a ship might look alike in a satellite image.

Kanjir et al. [4] had given a vast survey of vessel detection in the paper titled "Vessel detection and classification from spaceborne optical images: A literature survey." In this article, they have analyzed 119 articles to perform the detailed literature survey on vessel detection. The major areas of research highlighted by the paper include the object classification problem from satellite imagery and various factors that must be considered by performing the object detection. The article also highlights the opportunity of different types of satellite imagery with varied resolutions have a wide scope of research. Considering this, our proposed solution integrates deep learning approach for an effective object detection and classification of ships from Satellite Imagery.

Chen et al. [5] in the article "Deep learning for autonomous ship-oriented small ship detection" has proposed deep learning methods and its comparisons with various state-of-the-art methods. However, the image dataset repository they used are not from satellite imagery and they are mainly based on radar images. In satellite imagery, we get the actual images of the objects, while radar sends electronic signals to analyze the objects. This makes our proposed work an improvement in the field of object detection and classification from satellite images.

Huang et al. [6] in the article titled "A remote sensing ship recognition using random forest" has explained about various machine-learning algorithms that could be applied for ship detection. Their work has proven that random forest based on mutual information along with feature extraction helps to identify various ship targets in a precise manner. Moving from standard machine algorithms to deep learning algorithm gives more room of us to improve the feature extraction and to identify the objects from satellite images in a more accurate manner.

Krizhevsky et al. [7] in the year 2012 introduced the ImageNet classification using deep convolutional neural networks which commenced record breaking era of convolutional neural networks. From them, convolutional neural networks and its variants have been widely used for image classification. This is detailed by Rawat et al. [8] in the article "Deep convolutional neural networks for image classification: A comprehensive review Neural Computation."

Li et al. [9] has detailed about various feature extraction methods and algorithms that can be used in optical remote sensing images. CNN-based algorithms for object classification from remote sensing images is given in an elaborate manner. Efficiency in ship detection is given to be a scope of research in this article using CNN methods. In our proposed system, accuracy of the ship detection has achieved noticeable accuracy.

Stofa et al. [10] has used a deep learning approach for ship detection in harbor areas for possible pirate threats using DenseNet architecture. Their region of interest and the application used is different from our approach. Our major area of focus is to identify the ships that spill oil in the vast sea region. So, in this paper we have come out with an optimized deep convolutional neural network application, which performs the feature extraction. This also identifies and classifies whether there is a ship in the satellite imagery obtained with bounding boxes for fast and accurate identification. Let us dive deeper now. BERT is used for general purpose language understanding task. This article has acclaimed of retaining the capabilities of language understanding of BERT by 97%, reducing the size by 40% and 60% faster than the original.

3 Proposed Solution

Proposed solution in this article has used deep convolution neural networks to accurately identify and detect the ships from the satellite imagery. The hyper parameter settings chosen in the proposed solution are finalized after n of trials. The activation functions, filters used, and optimization algorithms used have been analyzed in a critical way in various combinations. Evaluation metrics and comparison of the state-of-the-art methods are described in the next section. The major domain used for building the system includes Python 3, which is the programming language used for the development of the ship detection model and ship detection system. Jupyter Notebook GUI has been used in order to utilize the tools like TensorFlow and Keras for faster processing with Anaconda 3 distribution. Various state-of-the-art models were tested using the same domain and same dataset.

3.1 Dataset Description

It is important to mention the dataset details used in the work conducted. We have taken one of the most popular open source and widely available dataset in Kaggle.com [11]. There are so many datasets available all over the Internet sources. However, we have chosen this based on the simplicity and reliability of data. This dataset has varied set of satellite images from markets, agriculture, defense and intelligence, energy, to finance. Moreover, this is a dataset created for identification of ships, which provides 4000 80×80 RGB-labeled images [11]. These images of earth are taken by the Planet Scope operated by Planet, which can provide images with almost 3 m per pixel resolution. Class labels are given as 1 and 0 with "ship" and "no-ship" label, respectively. There are 1000 images in "ship" class and 3000 images in "no-ship" class, which contains landcover areas, partial ship areas, and mislabeled ones by other previously trained model as shown in Table 1.

3.2 Proposed Architecture

The proposed architecture uses satellite images as input, and it is passed through various convolutional layers, max pooling layers, and dense layers before correctly identifying and classifying the ship as shown in the block diagram Fig. 1. The convolutions and pooling does the major job of feature extraction from the satellite images with the help of activation functions like Relu, tanH, leaky Relu, sigmoid, and softmax functions [12]. They correctly identify the decision boundary of ship in the satellite images. Various filters used in the convolutional network architecture also help to identify the features. However, the hidden layers are employed in the architecture handle. Various factors have been considered while building the model.

Table 1 Dataset details

Class label	Label name	Images
1	Ship	
0	No-ship	

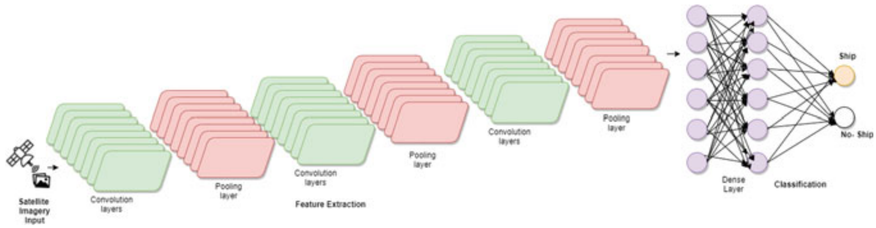


Fig. 1 Proposed architecture

The optimization functions played a major role. The optimization functions like Adam, RMSprop, and Stochastic gradient descent [13]. From among all these, Adam optimizer with a combination of binary cross entropy loss function gave the best results for our dataset. Binary cross entropy is chosen as our problem related to a binary classification of "ship" and "no-ship." It is very significant to efficiently choose the hyperparameters and learning rate of optimization algorithms. Sigmoid function is used as the activation function for the final layer, which gives a result between 0 and 1. Softmax can be replaced for sigmoid, if a percentage output is preferred. Callback functionality in tensor flow is used to tune the Keras model behavior while training, testing, and validation process [14].

Table 2 Hyperparameters of final model

Variables	Values
Convolution layer	3
Pooling layer	Max pooling
Activation function	ReLU
Batch size	32
Optimizer	Adam

4 Evaluations and Results

4.1 Setup

In this paper, the ship detection model was based on the convolutional neural network architecture. Python was used as the programming language to develop the deep learning model. TensorFlow and Keras were the API used to program the algorithm [15]. As shown in Table 1, the different parameters and variables are adjusted to best fit the model. The batch size is fixed and is set to 32 to get the best outcome when training the model. The model was trained using graphics-processing unit which enabled the model to train faster.

4.2 Optimizer

While utilizing the convolutional neural network architecture, different optimizer was used such as Adam, Adamax, RMSpropto, and SGD to get the best evaluation accuracy [16]. As shown in Table 2, most of the optimizer were significantly close in term of accuracy. However, the best optimizer for classifying the ship model is the Adam optimizer with an accuracy of 98.5%. Most of the optimizers used adaptive learning rate except for the SGD optimizer, which had the default learning rate of 0.01. The SGD optimizer was also the least performing optimizer with an accuracy of 89.25%. The batch size and epoch were fixed for all the optim**izers.**

4.3 Dataset Split

The splitting of the dataset was done to test the outcome of the accuracy of the model. The dataset is split into two cases in order to pick the best result for training the model. As seen in Table 3, the ratio of the splits between the training and validation are 70:30 and 80:20. The result indicates that having the dataset split between 70:30 showed a slight improvement over the 80:20. The hyper parameter when testing the split was fixed with a batch size of 32 and an epoch of 8.

Table 3 Optimizer's comparison

Optimizer	Batch size	Number of epoch	Evaluation accuracy (%)
Adam	32	8	98.5
Adamax	32	8	96.5833
RMSprop	32	8	97.5833
SGD	32	8	89.25

As shown in Fig. 2, the result of the confusion matrix shows the validation of the classified images. The total amount of validated images is 1200. These images are binary classified as "ship" and "no-ship." The result shows that the model classified a total of 280 true positive images of "ship" and a total of 7 false positive of "ship." Moreover, it classified a total of 902 true negative and a total of 11 false positive of "no-ship." The figure shows that the model is able to highly predict the correct images.

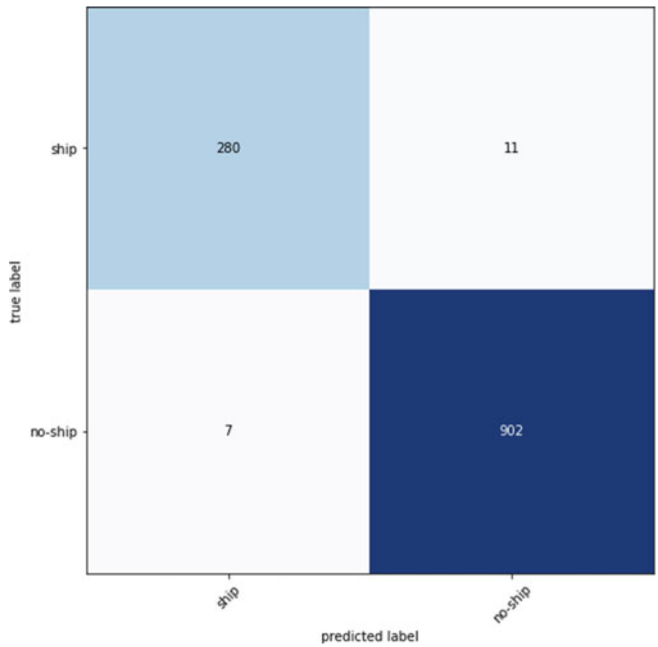


Fig. 2 Confusion matrix of 70:30 split of validation images

Table 4 Dataset split comparison

Training images	Validation images	Batch size	Number of epoch	Evaluation accuracy (%)
2800	1200	32	8	98.5
3200	800	32	8	97.87

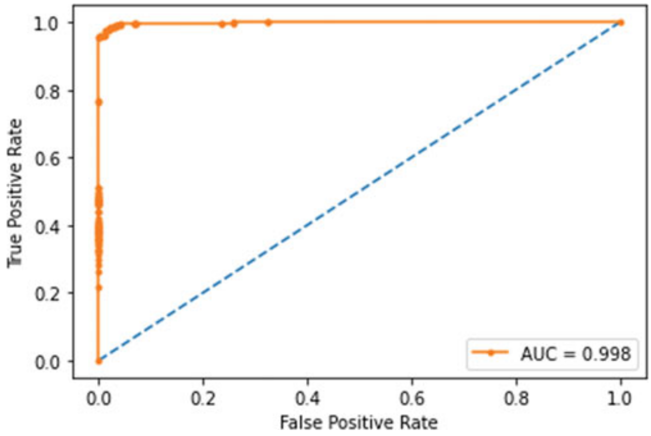


Fig. 3 AUC curve for ship model

4.4 Activation Function

To get the best evaluation accuracy for the ship model, different activation function was used like ReLU, Leaky ReLU, and TanH. Table 4 shows the different activation function used to get the best evaluation accuracy for the model. The result shows that ReLU was the best activation function when training the model which achieved an evaluation score of 98.5%, though the accuracy between the three different activation function was slightly close due to the fixed hyperparameter of the batch size and epoch [17] (Fig. 3).

As shown in Fig. 2, ReLU function was used to get the best result. The AUC curve of the ship model has a high score which is able to correctly classify the true positive "ship" and the true negative "no-ship."

4.5 Architecture Comparison

Deep learning architecture is used to build and solve different types of problems [18]. The convolutional neural network architecture is compared to the DenseNet architecture [10]. As shown in Table 5, both of the architectures used the same

Table 5 Activation function evaluation comparison

Activation function	Batch size	Number of epoch	Evaluation accuracy (%)
ReLU	32	8	98.5
Leaky ReLU	32	8	97.4167
Tanh	32	8	96.3333

Table 6 Architecture comparison between convolutional neural network and DenseNet

Architecture	Optimizer	Batch size	Number of epochs	Evaluation accuracy (%)
Convolutional neural network	Adam	32	8	98.5
DenseNet	Adam	16	2	98.4375

optimizer to get the best evaluation accuracy. However, the hyperparameters of the batch size and epoch are different. Both evaluation accuracies are close, though the convolutional neural network is ahead as the architecture used showed the better result. In addition, the convolutional neural network used adaptive learning rate to get the best result, while the DenseNet used fixed learning rate. Furthermore, the DenseNet showed lower evaluation accuracy result which is 89.6875% when using a learning rate of 0.01. Moreover, the convolutional neural network was trained with a graphics-processing unit while the DenseNet was **not** (Table 6).

5 Conclusion and Future Work

Marine contamination at the Arabian Gulf has been a subject matter of extensive international and regional fear. From the concluded underlying analysis, regional and international standards have the possibility of considerably decreasing the incidence of pollution events. This research has proposed a novel CNN-based ship recognition structure for high-resolution optical remote sensing images. To adequately train the proposed S-CNN model, we accumulated positive samples with several kinds, dimensions, and structures from more than one thousand high-resolution remote sensing images and a huge number of negative samples. To summarize, the ship detection system revolves around deep learning networks that are built to deliver a state-of-the-art solution for detecting vessels in the oceans. This will aid in lessening the influence of oil spills in our marine ecosystem by utilizing high-resolution satellite images to train the model. Furthermore, this study has been accomplished with an accuracy of 98% by using three convolution layers with ReLU activation function and max pooling. The system has been tested and is expected to be well preserved by its end users, and to governmental entities, thus reducing cost and effort on our countries.

As part of the future work, the following study shall include several different technologies to have a full workflow of the scenario. It is critical to detect the ship that is responsible of an oil spill, by monitoring and tracking the ship movement and position inside the ocean by utilizing the Automatic Identification System (AIS). The AIS are devices that are assembled on vessels to avoid collisions and uses transmitter systems to track its location. Moreover, integrating GNOME tool into the ship detection system will be of a huge benefit, as it will allow us to predict the exact route of the spread of oil spill. This can be done by determining the movement of the wind, currents, and other process. Finally, a ship traffic detection system will be created to track incoming and outgoing of ships. The execution of this system will assist in lowering traffics and unintentional vessel collusion, thus diminishing oil spill.

References

1. Aldosari KR (2019) The maritime commons: digital repository of the World Maritime University. World Maritime University, Malmo
2. Comack D (1999) Response to marine oil pollution. Kluwer Academic Publishers, Boston
3. Society LO (2011) Tanker technology limitation of double hulls. Sointula
4. Kanjir U, Greidanus H, Oštir K (2017) Vessel detection and classification from spaceborne optical images: a literature survey. Rem Sens Environ, p 26
5. Chen Z, Chen D, Zhang Y, Cheng X, Zhang M, Wu C (2020) Deep learning for autonomous ship-oriented small ship detection. Safety Science
6. Huang S, Xu H, Xia X (2015) A remote sensing ship recognition using random forest. In: 4th International conference on information science and cloud computing, Wuhan
7. Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. In: Advances in neural information processing systems 25, Toronto
8. Rawat W, Wang Z (2017) Deep convolutional neural networks for image classification: a comprehensive review. Neural Comput
9. Bo LI, Xiaoyang XI, Xingxing WE, Wenting TA (2021) Ship detection and classification from optical remote sensing images: a survey. Chinese J Aeuronaut 34(3):145–163
10. Stofa MM, Zulkifley MA, Zaki SZ (2020) A deep learning approach to ship detection using satellite imagery. In: IOP conference series earth and environmental science, Malaysia
11. rhammell, Kaggle (2018) Available: https://www.kaggle.com/rhammell/ships-in-satellite-imagery
12. Agostinelli F, Hoffman M, Sadowski P, Baldi P (2015) Learning activation functions to improve deep neural networks. ICLR
13. Thoma M (2017) Analysis and optimization of convolutional neural network architectures, Karlsruhe Institute of Technology
14. TensorFlow, September 2021. Available: https://www.tensorflow.org/api_docs/python/tf/keras/callbacks/Callback
15. Ashraf M, Ahmad SM, Ganai NA, Shah RA, Zaman M, Khan SA, Shah AA (2021) Prediction of cardiovascular disease through cutting-edge deep learning technologies: an empirical study based on Tensorflow, Pytorch and Keras. In: International conference on innovative computing and communication
16. Kumar R, Singh RC, Kant S (2021) Dorsal hand vein-biometric recognition using convolution neural network. In: International conference on innovative computing and communication

17. Chakraborty N, Dan A, Chakraborty A, Neogy S (2020) Effect of dropout and batch normalization in siamese network for face recognition. In: International conference on innovative computing and communication
18. Madhavan S (2017) Deep learning architectures. IBM Developer

Face Mask Detector Using Convolutional Neural Networks

Rajendrani Mukherjee, Akash Narain Panday, Sanjukta Nandy, Sushmit Ghosh, Supratim Bhattacharya, Apurba Dey, and Saumadip Dey Choudhury

Abstract A convolutional neural network (CNN) has one or more layers and is mainly used for image processing, classification, segmentation. CNN is commonly used for satellite image capturing or classifying hand written letters and digits. In this particular project, a convolutional neural network is trained to predict whether a person is wearing a mask or not. The training is done by using a set of masked and unmasked images which constitutes the training data. The performance of the trained model is evaluated on the test dataset, and the accuracy of the prediction is observed.

Keywords COVID-19 · Convolutional neural networks · Image processing · Mask detection

1 Introduction

COVID-19 is a communicable disease caused by SARS-CoV-2. The disease has spread worldwide, surfacing as an on-going pandemic [1, 2]. Symptoms of COVID-19 consist of fever, cough, headache, fatigue, breathing difficulties, and loss of smell and taste [3]. As of July 7, 2021, 3,995,565 deaths have been reported for 184,710,938 cases (2.2%) [4].

COVID-19 mortality can be viewed from various angles. Belgium and some other countries include deaths from COVID-19, no matter whether the person was tested or not. Appraisal of death rate in pre-pandemic and during pandemic era showed some mismatch as there are many deaths which are not because of COVID-19 only [5]. Considering these entire scenario, worldwide COVID-19 deaths are normally ranging from 7 to 13 million [6]. RTPCR testing or RTLAMP from swab testing are standard methodologies which are used for COVID diagnosis. Preventive measures

R. Mukherjee (✉) · A. Narain Panday · S. Nandy · S. Ghosh · S. Bhattacharya · A. Dey · S. D. Choudhury
Department of Computer Science and Engineering, University of Engineering and Management, Kolkata, India
e-mail: rani.mukherj@gmail.com

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_33

include physical or social distancing, quarantining, cleaning hands, etc. Face masks are highly effective in preventing transmission.

In this paper, a mask detection system is proposed to enforce one of the most important preventive measures to contain the spread of the corona virus, i.e., wearing a mask. The detection functionality is implemented by training a convolutional neural network with a set of masked and unmasked images. The performance of the trained CNN model is measured by a number of metrics such as training accuracy, training loss, validation accuracy and validation loss. The OpenCV library is used for capturing live video feed. The trained CNN model is then applied on the detected face to predict whether it is masked or not.

The paper is divided into following sections: Introduction is in first section, literature survey is in second, the face mask detector is discussed briefly in Sect. 3, fourth section states the result of our experiments and conclusion, and future scope is the last section of research paper.

2 Related Work

Usage of machine learning approaches for practical problem solving is age old [7–9].

Turk and Pentland performed PCA where each face could be represented by numerical weights [10]. Fisherfaces reduced classification error rates from 24 to 7% [7]. In 2001, Viola and Jones avoided features such as eyes, ears, nose, mouth, color, and skin-tone detection and focused on the technique of boosting [11] to analyze faces.

David H. Hubel and Torsten Wiesel received the Nobel Prize in Physiology or Medicine in 1981 for identifying many neurons which react to stimulus located in limited visual region [12].

Many researchers have utilized convolutional neural networks (CNN) [13] for healthcare analytics. LeCun et al. designed LeNet-5 architecture [14] which is very novel in design. Apart from some fully connected layers and activation function, the study introduced pooling layers.

3 Implementation

The face mask detection method is implemented by training a convolutional neural network to distinguish between masked and unmasked images. The dataset used for training of CNN constitutes of two sets—masked images and unmasked images, each of which contains over 600 images (Fig. 1).

The OpenCV library is used for data pre-processing. All the images present in the training dataset are converted to gray scale (Fig. 2) (because face detection is faster in gray-scaled images), resized to 100 * 100 pixel, and are stored in an array. Each image is given a label: 0 for mask and 1 for unmasked. The data are normalized by

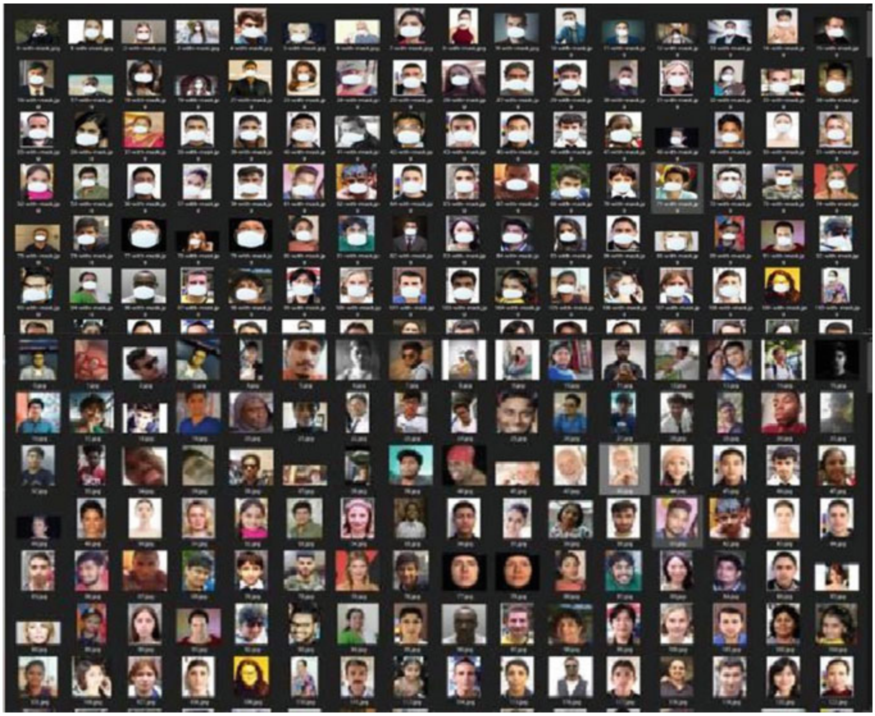


Fig. 1 Dataset of masked images and unmasked images

Fig. 2 Structure of data pre-processing

dividing the array with 255 (since a pixel can have a maximum value of 255) so that each pixel of every image has a value within the range of 0 and 1. It is then converted to a 4D array and is saved.

After all the images of the training dataset are processed, the convolutional neural network is set up. The sequential model of the Keras library is used to connect all the layers of our neural network (Fig. 3). The first layer is a convolutional layer of 200 kernels of size 3 * 3 followed by ReLU layer and max pooling layer. The second layer is also a convolutional layer of 100 kernels of size 3 * 3 followed by a ReLU and max pooling layer. The convolutions are flattened by using a flatten layer (Fig. 4) which helps in simplifying each input image into one dimensional array. A

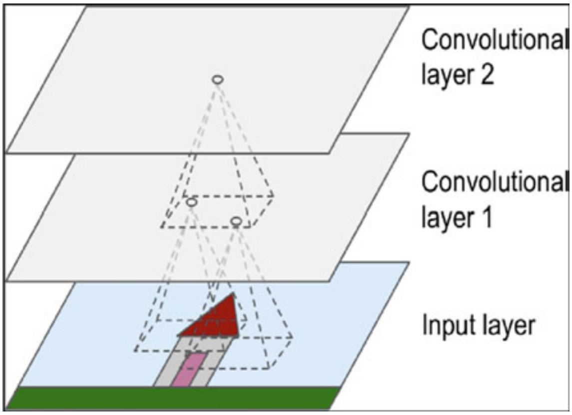


Fig. 3 Two convolutional layers with rectangular local receptive fields

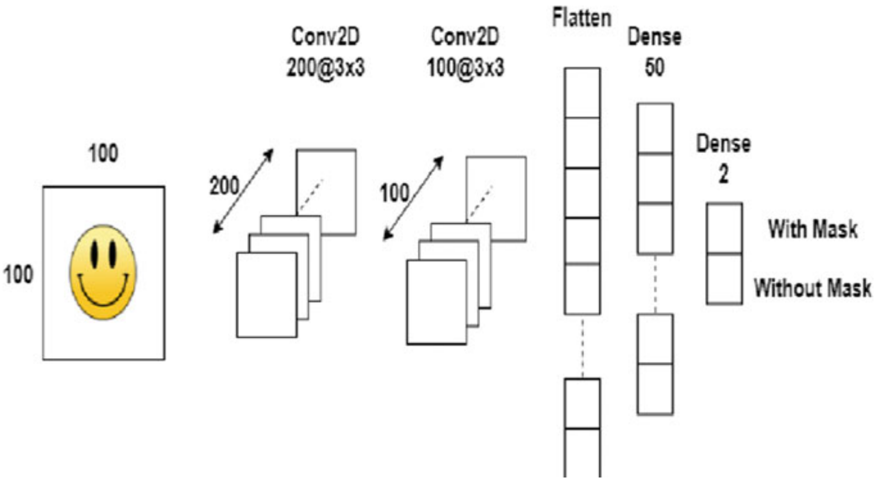


Fig. 4 Schema of the convolutional neural network

dropout layer is also included to reduce over-fitting. Finally, two dense layers are created constituting of 50 and 2 neurons, respectively (two neurons in latter because our output consists of two categories—with mask and without mask). The activation function of the former dense layer is ReLU while that of the latter dense layer is softmax.

After setting up the neural network, it is then compiled. The 'categorical_crossentropy' loss function is used since there are two categories ('with mask' and 'without mask'). The optimizer used is Adam optimizer which means that the model is trained using Adam's optimization algorithm. Stochastic gradient descent could have also been used, but it was observed that Adam's optimization algorithm was providing a better accuracy. The dataset was split into the training set and the testing set in the ratio 9:1 meaning that 90% of the total available data was used for training and the remaining 10% was used for testing. The model was then trained with the number of epochs being set to 20 meaning that the training dataset was passed forward and backward through the neural network 20 times. The model for each epoch was saved, and the validation loss for each epoch was monitored. The epoch that gave the best model was saved for detection of mask. Python's Matplotlib library was used to display the graph for validation loss and training loss as well as the graph for training accuracy and validation accuracy.

For face detection, the OpenCV library is used for capturing live video feed, and the Haar cascade frontal face classifier is used for detecting faces. The trained CNN model is then applied on the detected face to predict whether it is masked or not. A red rectangle box appears around the detected face with the label 'no mask' in case the person is unmasked while a green rectangle box appears with the label 'masked' in case the person is masked.

4 Results and Findings

After 20 epochs, the training accuracy was observed to be 98.14% (Fig. 6) and the training loss was 4.18% while the validation accuracy was observed to be 96.48% and the validation loss was 16.11%. A graph is plotted with the number of epochs on the *X*-axis and the losses (training loss and validation loss) on the *Y*-axis (Fig. 5).

From the graph, it can be inferred that there is over-fitting of the data (since for a particular epoch the blue line is above the red line). Hence, our model is not perfectly trained. However, it is more than enough for the detection of mask under ideal lightning conditions. The face detection algorithm is able to detect multiple faces which were expected. The algorithm is even able to detect faces even if it is slightly titled at an angle. However, the face detection failed when the concerned person's head is at an angle such that it one of the notable facial features such as the person's eyes or nose is not visible to the camera. Prior testing showed that the model struggled to detect faces in extreme lightning conditions. The tests were carried out on a 480px resolution camera and a higher resolution camera; particularly, a HD

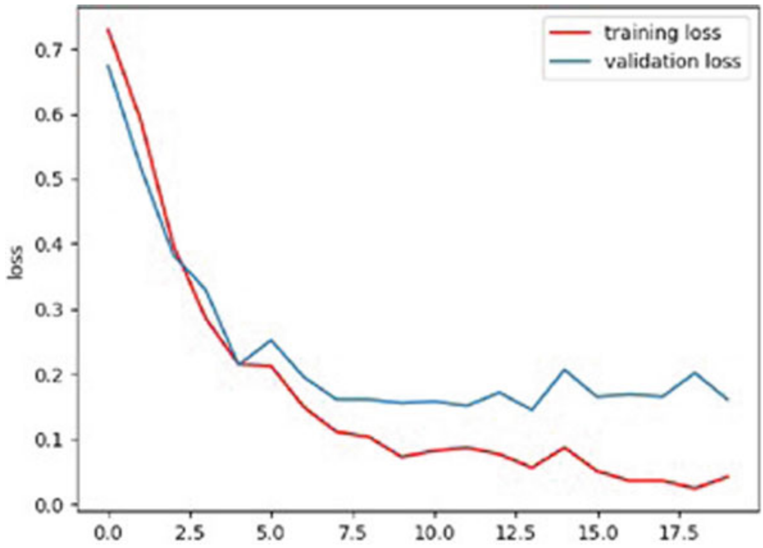


Fig. 5 Graph depicting the losses (training loss and validation loss)

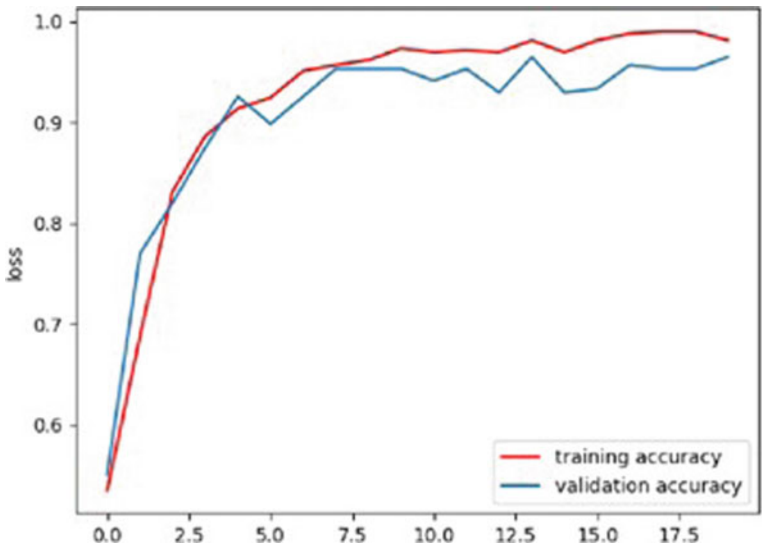


Fig. 6 Graph depicting the accuracy (training accuracy and validation accuracy)

Fig. 7 Demonstration of mask detection system while unmasked and masked

camera is expected to provide much better results even in cases of intense or low lightings.

After the face detection, the detection of mask was more or less consistent. Several types of masks were used for testing. A variety of colored masks were used as well. The algorithm detected the mask quite efficiently every time in spite of the variations. The labels for 'masked' and 'unmasked' also appeared on the video feed efficiently whenever the user was masked and unmasked (Fig. 7).

5 Conclusion

The COVID-19 pandemic will continue to spread, if necessary preventions are not taken. Wearing a mask is one of the most important and efficient preventive measures in conquering this on-going pandemic. Our project is not only an attempt to encourage wearing of masks but to enforce it as well. Face detection is a common tool in today's technology and is used in variety of applications such as DSLR cameras, smart phones, and surveillance systems. In our project, we have integrated the mask detection and the face detection system to determine whether a person is wearing a mask or not. Our project has promising hardware applications. The code can be implemented in surveillance systems such as CCTV cameras. A shop owner has every right to deny service to a person who is not wearing a mask since the person carries a major risk of infecting the other customers as well as the owner. However, manually denying entry every time to an unmasked person can be troublesome. The person may even end up arguing with the owner. However, if there is an automated system that can segregate those who are masked from those who are unmasked, then it will ease a lot of the shopkeeper's troubles. The objective of this project is to achieve exactly that.

References

1. Page J, Hinshaw D, McKay B (2021) In hunt for Covid-19 origin, patient zero points to Second Wuhan Market. Wall Street J
2. Scientific Brief: "SARS-CoV-2 Transmission", Centers for Disease Control and Prevention
3. Agyeman AA, Chin KL, Landersdorfer CB, Liew D, Ofori-Asenso R (2020) Smell and taste dysfunction in patients with COVID-19: a systematic review and meta-analysis. Mayo Clinic
4. Scientific Brief, "COVID-19 Dashboard", Center for Systems Science and Engineering (CSSE) at Johns Hopkins University
5. Lazzerini M, Putoto G (2020) COVID-19 in Italy: momentous decisions and many uncertainties. The Lancet
6. Report of Institute for Health Metrics and Evaluation, "COVID-19 Projections"
7. Belhumeur P, Hespanha J, Kriegman D (1997) Eigenfaces vs. fisherfaces: recognition using class specific linear projection. IEEE Trans Pattern Anal Mach Intell
8. Mukherjee R, Sridhar Patnaik K (2019) Introducing a fuzzy model for cost cognizant software test case prioritization. In: IEEE TENCON 2019: recent advances in program analysis and software testing (RAPAST), pp 502–507
9. Mukherjee R, Sridhar Patnaik K (2019) Prioritizing JUnit test cases without coverage information: an optimization heuristics based approach. IEEE Access 7(1):78092–78107. https://doi.org/10.1109/ACCESS.2019.2922387
10. Turk MA, Pentland AP (1991) Face recognition using eigenfaces. In: Proceedings of CVPR'91, computer vision and pattern recognition
11. Viola P, Jones M (2001) Rapid object detection using a boosted cascade of simple features
12. Hubel DH, Wiesel TN (1962) Receptive fields, binocular interaction and functional archtecture in the cat's visual cortex. J Physiol 160(45):106–154
13. Fukushima K (1980) Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biol Cybern 36:193–202
14. Lecun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proc IEEE 86(11):2278–2324. https://doi.org/10.1109/5.726791

A Secure and Reliable E-Health Data Transmission and Remote Patient Monitoring Over an Internet of Things Framework: A Step Forward to Mitigate Community Spread of Coronavirus

Bazila Parveiz, Ravinder Pal Singh, and Monika Mehra

Abstract The Internet of Things (IoT) has revolutionized the ways; the physical world is connected to the cloud for real-time data dissemination through embedded sensors and microcontrollers. IoT plays an important role in almost every sphere of the world, may it be physical sensor connection to the cloud, the Structural Health Monitoring Systems (SHMS), Smart Homes, Health care, etc. Healthcare Internet of Things (H-IoT) has taken the healthcare sector to the next level by incorporating remote patient monitoring and diagnosis, Robotic surgeries, patient's vital data monitoring in real time, etc. This paper presents a novel and a simple technique of remotely monitoring patients suffering from highly contagious diseases like the Corona Virus, thereby reducing the direct patient–doctor physical contact and ensuring the social distance. The patients' medical data is acquired and end-to-end encryption is done on the data to ensure no loss of the data between the transmitting end and the receiving end. The designed system is based on the Node_Mcu microcontroller platform. The sensor data is acquired and processed using the Arduino-Integrated Development Environment (IDE) and further predictions regarding the patient's health are performed in the MATLAB 2019a computation software.

Keywords Healthcare IoT · E-Health · Corona virus monitoring · Remote patient monitoring system · Health care data encryption

1 Introduction

IOT build solutions for numerous applications like agriculture, water, management, industry, health care has come into being. IOT is a constructed chain system of the information transferred from and to every inter-related 'smart' device via Internet.

B. Parveiz (✉) · R. P. Singh · M. Mehra
Department of ECE Engineering, RIMT University, Mandi Gobindgarh, Punjab, India
e-mail: bazilaparveiz25@gmail.com

M. Mehra
e-mail: monika.mehra@rimt.ac.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_34

Health maintenance applications in IOT can provide extensive care to a patient in many cases [1]. Rise in infectious diseases has led to rise in use of smart health maintenance, to prevent the spread of the most contagious viruses like the novel corona virus, which is done by tracking data from bedside of patient and diagnosing in real time [2, 3] by which patient care is improved to another level. Effective health care depends on speed and accuracy. Without physical communication, doctors can give remote diagnosis, providing quality and quick medical supervision to rural and urban areas. Using sensors and Wi-Fi, getting vital details becomes straightforward. Through a single application, we can keep track of IOT data. Thus, IOT allows staff to do their jobs perfectly. With all this, two main things come into being – Sensor data security and efficiency of local and global communication. As there is increase in dependency and facility of IOT, one major problem comes into being is the security of critical patient data.

This paper proposes a secure and reliable E-Health data transmission scheme considering real-time patient data dissemination to doctors and security of data through end-to-end encryption. The proposed scheme focusses more on the real-time data security. The encryption is performed on the smart sensor side on hardware and the decryption is done on the server's side where the data is visible to the concerned doctor only. The medium of data transport from the sensor side to the server side, i.e. an Internet link experiences encrypted data. Hence, there are minimal chances of data theft and intersection. The overall system is implemented over a ThingSpeak IoT analytics platform. Another major feature of the proposed idea is to avoid direct physical contact between the patients and the concerned healthcare workers to prevent spread of highly contagious infection such as coronavirus. The main motivation behind this work is to implement a remote monitoring healthcare system, which may help in mitigating the community spread of corona virus by minimizing the patient–doctor physical appointments. Since the system operates in real time, it is highly challenging to compensate for the transmission delay and the packet loss during real-time vital data transmission over an IoT framework. This problem will affect the data integrity, especially for real-time COVID-19-infected patient's vital information such as the body temperature, the oxygen saturation levels and the heart rate. Inaccurate vital data might lead to misdiagnosis. IOT in terms of health management has played a key role. Health Internet of Things (H-IOT) deals with the remote monitoring of critical care patients by connecting them to various biomedical sensors for real-time biological vital acquisition and actuators to provide timely medical support. Smart health maintenance is the most challenging and critical application of IOT And H-IOT [4]. The Internet of Things (IoT) is giant network of sensor networks, processors, actuators and Internet-connected devices. Internet of Things provides the capability to keep track on people, prototype, apparatus, or assistant animals and analysing the data collected. IOT has unfolded as one of the revolutionary technologies [5] that have worldly, economic and social impact. Massive resources in long-term cases are to be spend eventually making it difficult to provide with qualified long-term health management. Many IoT-enabled embedded devices have been developed for this purpose, such as in smart wearables, smart video cameras and fitness shoes. By 2025, *the number* of Internet-connected *medical* *devices* [4] is expected to reach unprecedently more than 50 billion. Numerous IOT-enabled devices exist and large number of applications are running in the cloud ubiquitously. Health maintenance group has to keep a close track of the patient's health since their condition changes continuously, which becomes impossible to record in real time. With this, e-Health provides [6, 7] the most critical application to patients, where there will be virtual patient–doctor appointments sitting anywhere in the world through leveraging the power of the Internet. Actuators provide timely medical support. Smart health maintenance is the most challenging and critical application of IOT. H-IOT [8, 9] collects the patient's data, analyses it and processes it to provide better diagnostic care for them. Swift increase in world population provides critical challenge to existing health maintenance and medical services. Spike in the medical support [10] has led to unexpected patient–doctor appointments.

2 Related Work

A number of studies have been carried out pertaining to the field of IoT in general and H-IoT in particular. A surveillance system for COVID-19 patients in self-quarantine to keep monitoring [8] of physiological data like SpO_2 and heart rate with track of the patients is proposed. For data accession and conveyance, wireless body sensors and a gateway are implemented. A framework for SPO_2level accessing [9] and austerity calculation and use of determination decision of being a COVID-19 patient is carried out. A Real-time Patient Health Assessment [10] and apprise through IoT is initiated and executed using ThingSpeak IoT cloud. A distinctive identification system is preferred for monitoring [11] latest infectious disease like coronavirus (COVID-19). Scouting and analysis are carried out and related the IoT-capacitated technologies [12] and applications used in contact tracing, screening and surveillance. This research presents accessing and registering device for heart rate, SPO_2 levels [13] and body temperature. An extensive survey have been conducted in [14] on the classification technique in Cloud and IoT-based health monitoring and diagnosis approach. The healthcare data acquired from any repository or by using sensors that collect real-time data are stored in tables for reference to the severity of the disease by any professional. Various types of applications of IoT and their applicability have been focussed upon in [15, 16] for use in the healthcare system. Expectation of transmission of patient healthcare data in real-time mode to doctors [13] and agencies by involvement of virtualization and fuzzy inference system in fog computing. For healthcare convergence, mobile device gateway, typically an integrated gateway [17], is created that approves heterogeneous devices. Facilitation of smart healthcare applications and services proposed a novel platform with machine to machine messaging, theorem-proving beacons [18] for consistent data management through data fusion and decision fusion. Security threat to the IoT things led to a multi-agent approach for continual threat detection [19] with the use of machine learning for prospective analyses: identifying security vulnerabilities to make predictions. Generation and aggregation of data is being carried out more than ever before whilst healthcare progress towards patient-directed [20] and analytic application. Discussion of patient-directed health care along with health data is accomplished. Constraints to IoT applications as of robustness, security, privacy, and reliability [21] leads to surveying of various healthcare solutions using IoT. The overall paper is organized as follows: Sect. 3 discusses the proposed system. Section 4 focusses on the methodology. In Sect. 5, the hardware implementation is presented. Firmware development for Node-MCU is discussed along with the flowchart in Sect. 6 and, finally in Sect. 7, the results are presented and discussed.

3 Proposed System

The proposed system helps mitigating the spread of any highly contagious disease such as a coronavirus by minimizing the physical contacts between the infected persons and the healthcare workers attending them [22]. The system makes use of the smart wearable medical sensors capable of providing real-time data. The sensors used are infrared temperature sensor module capable of measuring patient's body temperature; Max30102 from Spark Fun Electronics designed to measure the oxygen saturation levels (SPO_2) and the numbers of Beats per Minute (BPM), the Ad fruit's AD8232 Electrocardiogram (ECG) sensor and the DHT11 sensor for measuring the patient's body humidity. All the sensor data acquisition and calibration are coordinated and encrypted by the Node_Mcu Microcontroller board having onboard Wi-Fi module for Internet connectivity. The architecture of the overall proposed system is depicted in Fig. 1. The sensor data acquired is first encrypted using the bitwise encryption techniques. The encrypted data is then transmitted to the ThingSpeak IoT analytics platform for processing, analytics and future predictions. The encrypted data is retrieved from the ThingSpeak platform in MATLAB and decrypted using bitwise decryption techniques. The decrypted data and the actual sensor data show a high level of integrity, thereby disseminating the infected patient's vital data in real time to the doctors for timely and safe diagnosis.

4 Methodology Adopted

The implementation of the proposed E-Health system is accomplished in four phases: the medical sensor data acquisition using different connection protocols, the sensor data encryption on the core of ESP8266, transmission of the encrypted data to the ThingSpeak IoT analytics platform and finally the decryption of the retrieved data by the MATLAB Computing software. In addition to decryption, the MATLAB also performs analytics and future predictions on the retrieved patient's medical data. The detailed description of the four phases is given as under.

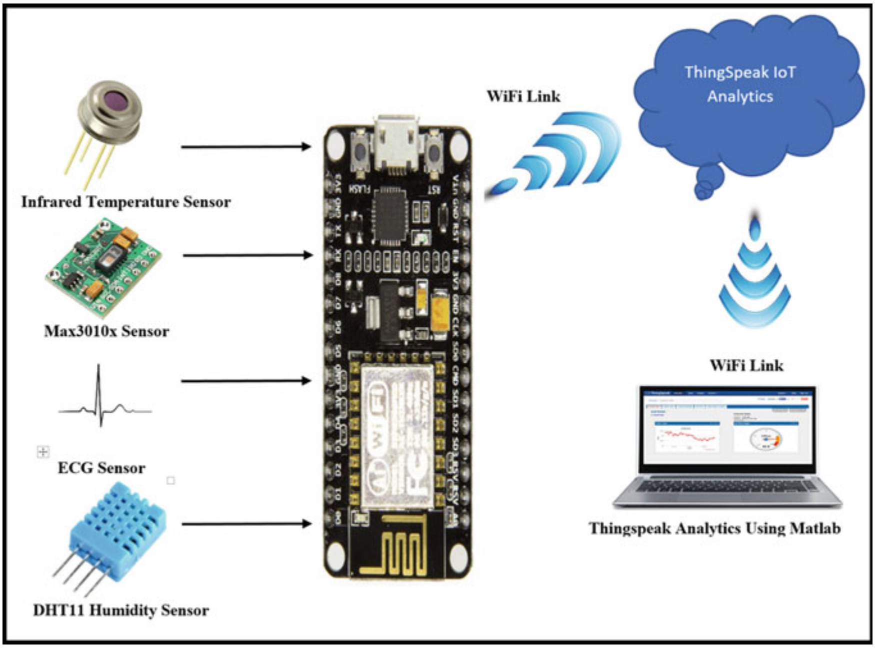


Fig. 1 Architecture of the proposed system

4.1 Sensor Data Acquisition

Different sensors are interfaced differently; the infrared temperature sensor and the Max30102 sensor are interfaced with the Node_Mcu via the Inter-Integrated Circuit (I^2C) protocol. The I^2C bus consists of one master device (Node_Mcu) and any number of slave devices (I^2C-enabled sensor devices). Each device on an I^2C bus is assigned a unique address so that the master device can choose with which device to communicate. The ECG sensor is an analogue sensor and is interfaced directly with the analogue input of the Node_Mcu board. The DHT11 sensor is a digital sensor and needs the digital input for interfacing with the microcontroller board.

4.2 Sensor Data Encryption

The patient's sensor data is encrypted using various logical bitwise encryption techniques such as bitwise shifting, bitwise complement, bitwise XOR and bitwise X-NOR. These operations are used in the process of encryption on the smart sensor side and decryption on the end user side. The sensor data is converted into 8-bit binary equivalent, and the bitwise encryption operations are performed on the binary equivalent of the sensor readings. The encrypted binary equivalent is converted to decimal

Fig. 2 Sensor data encryption technique

and transmitted to the ThingSpeak IoT cloud for storage, processing, analytics and future predictions. Various steps involved in the encryption of sensor data are shown in the Fig. 2.

4.3 Data Transmission to ThingSpeak

ThingSpeak is an open source analytics platform used for the development of the IoT applications. It is used for real-time data collection, analysis of data and visualizations of data through charts. Other important features of ThingSpeak include storage and retrieval of data from things using the Hypertext Transfer Protocol (HTTP) and Message Queuing Telemetry Transport (MQTT) over Internet or any other Local Area Network (LAN). The reason behind choosing ThingSpeak IoT platform for our work is that MATLAB has an integrated support for ThingSpeak to analyse and visualize data without licence. ThingSpeak aggregates the sensor data sent from the Node_Mcu, visualizes the data and performs analysis of live data streams on the cloud. This sensor data is stored for future reference and making predictions about the future health conditions of the patients suffering from persistent long-term diseases. With the MATLAB engine in ThingSpeak, it is easy to perform calibrations, develop Analytics, transform IoT data and build custom charts. The location widget in ThingSpeak helps in locating the patients whose data is being transmitted to the cloud, so that timely medical assistance can be provided before any catastrophic situation takes place.

4.4 Sensor Data Decryption

The integrated ThingSpeak support in MATLAB is used to retrieve the encrypted data from the IoT platform. The retrieved data is decrypted by following a technique opposite to that shown in Fig. 2. The retrieved medical data of the patient is converted to binary equivalent, and the corresponding bitwise decryption technique is applied. The binary equivalent of the decrypted sensor data is converted to decimal and presented to the concerned doctor for diagnosis. A MATLAB database stores the received sensor data. This data is further subjected to machine learning algorithms for future health condition predictions. Various steps involved in the data retrieval and decryption process are shown in Fig. 3.

Fig. 3 Sensor data retrieval and decryption technique

5 Hardware Implementation

The system design has been carried out with Node_Mcu as the main processing controller. Node_Mcu is a combination of firmware and hardware based around the ESP8266-12E module, which is a low power inexpensive Wi-Fi-enabled microchip with a full Transmission Control Protocol/Internet TCP/IP stack and microcontroller capabilities [23]. The board is powered by a 5-volt Universal Serial Bus (USB) connection. The Max30102 is an integrated pulse oximeter and a heart rate module. It has an inbuilt Light Emitting Diode (LED) and a photo detector for measurements. The module comes with an integrated I^2C interface and can be used with any host controller supporting I^2C communication. The MLX90614 temperature sensor is a contactless sensor. Temperature measurements become easy and safe without any physical contact with patient. This sensor also comes with an integrated I^2C interface. The AD8232 ECG sensor is a three-electrode single lead breakout board, which measures the electrical signals generated, when the human heart contracts and expands during blood pumping and circulation. The placement of the electrodes along with the colour codes is shown in Fig. 4. Different electrode placements will result in different shapes of the ECG signals as different muscles are measured.

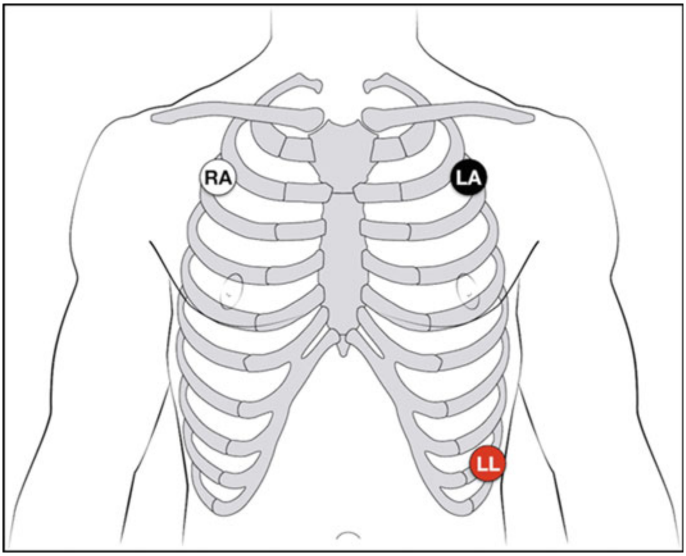


Fig. 4 Placement of ECG electrodes

The DHT11 is a capacitive humidity sensor used for the patient's body humidity. It also has a thermistor for temperature measurements, which is less sensitive for body temperature measurements and hence is discarded.

6 Software Design

The development of the software is done in two environments: Arduino and MATLAB. The Arduino-Integrated Development Environment (IDE) is used to write firmware for the Node_Mcu microcontroller. This firmware helps in sensor data acquisition, Wi-Fi connection management, encryption of sensor data and sending the encrypted sensor data to ThingSpeak IoT platform for analytics, processing and future references. The complete software design for the Node_Mcu side is illustrated with the help of a flowchart shown in Fig. 5. In MATLAB environment, the encrypted sensor data from ThingSpeak is retrieved, decrypted and analysed for future predictions. The analysed data is then forwarded to the medical specialists for timely remote diagnosis. The MATLAB firmware design flow is shown in Fig. 6.

7 Results

In this section, we discuss the results. The implemented system captures the most vital parameters of the patient, such as body temperature, oxygen saturation levels, body humidity and the real-time heart rate. In Fig. 7, the patient's body temperature is obtained as a continuous function of time. This temperature data is encrypted using the bitwise encryption technique and transmitted over the ThingSpeak IoT platform. Figure 8 represents the encrypted data transmitted over an IoT network. In Fig. 9, the patient's body humidity is obtained as a continuous function of time. This humidity data is encrypted using the bitwise encryption technique and transmitted over the ThingSpeak IoT platform. Figure 10 represents the encrypted data transmitted over an IoT network. In Fig. 11, the patient's oxygen saturation level (SPO_2) is obtained as a continuous function of time. This SPo2 data is encrypted using the bitwise encryption technique and transmitted over the ThingSpeak IoT platform. Figure 12 represents the encrypted data transmitted over an IoT network. In Fig. 13, the patient's real-time heart rate is obtained as a continuous function of time. This heart rate data is encrypted using the bitwise encryption technique and transmitted over the ThingSpeak IoT platform. Figure 14 represents the encrypted data transmitted over an IoT network. The encrypted vitals data in all the above-mentioned four cases is received through a MATALB terminal accessible to the concerned doctor on the other side. Future predictions and further processing on the data are done in MATLAB to predict the vulnerability of the diseases infecting a particular patient.

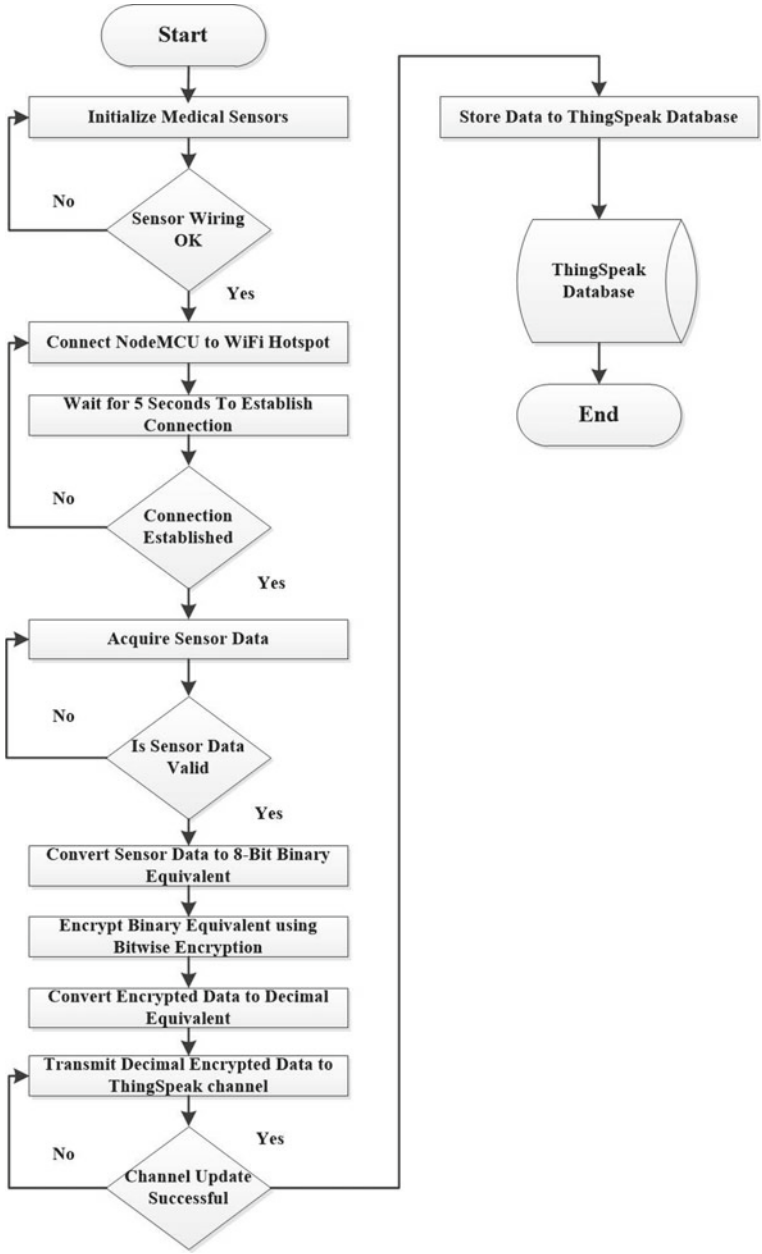


Fig. 5 Flowchart of firmware for Node-MCU microcontroller

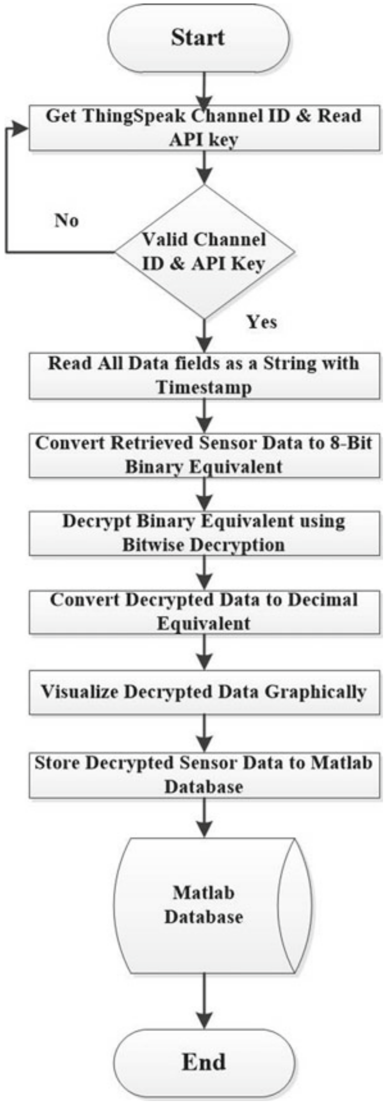


Fig. 6 Flowchart of software design using MATLAB

8 Conclusion

The proposed system was successfully implemented and tested. The real-time medical data obtained from the sensors was very well calibrated and presented in an easily and understandable format. In this paper, we mainly focus on the security of the data rather than the acquisition of data. The encryption techniques employed provided real-life results resembling to the standard medical data. The ThingSpeak IoT platform is used to aggregate, visualize, and analyse live data streams of the

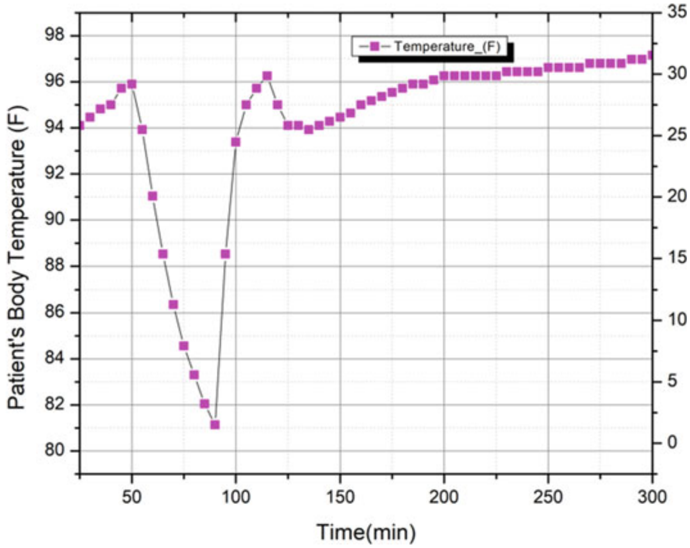


Fig. 7 Patient's body temperature

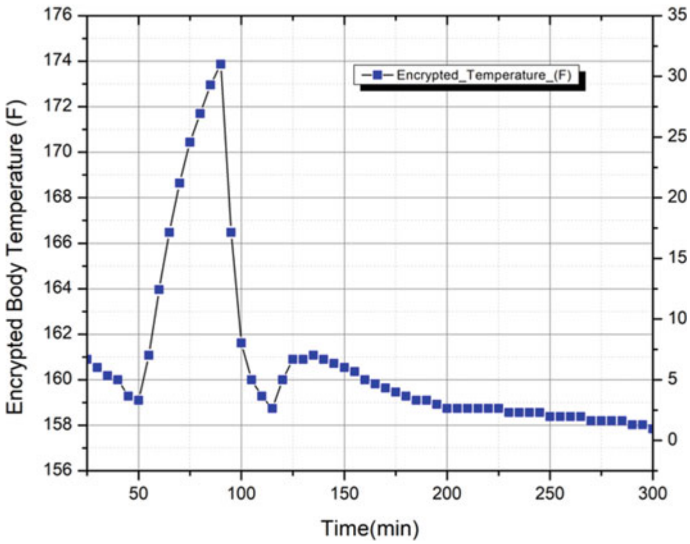


Fig. 8 Encrypted patient's body temperature

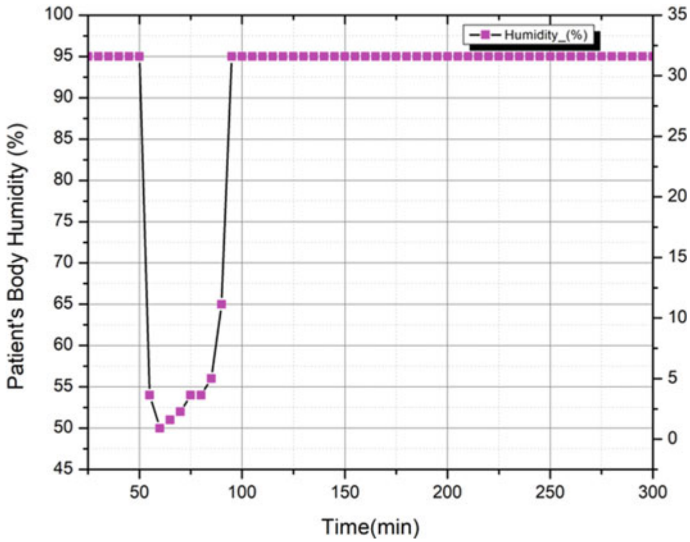


Fig. 9 Patient's body humidity

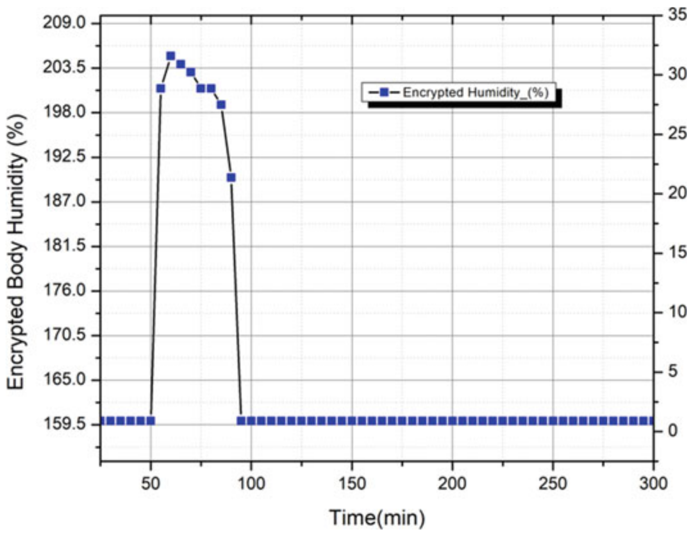


Fig. 10 Encrypted patient's body humidity

patient's medical data obtained from the sensors in the cloud. This platform has an amazing feature of integration with MATLAB that helps in easy retrieval of data from the doctor's side. Various bitwise encryption schemes tested in this work includes 1's complement, 2's complement, exclusive OR and exclusive NOR.

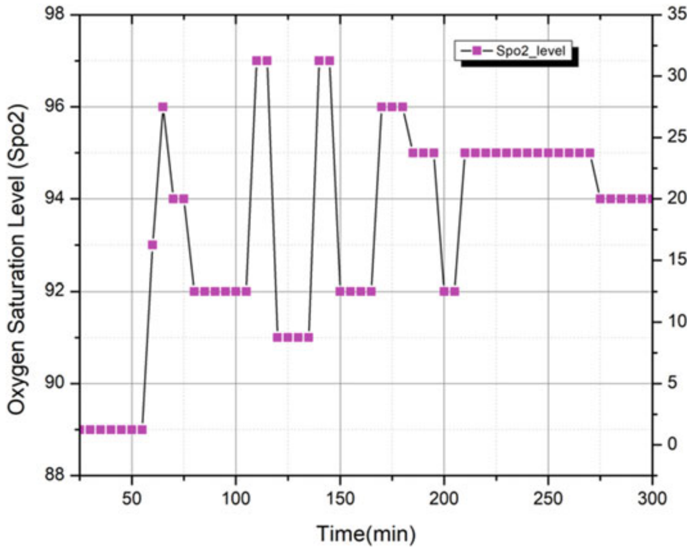


Fig. 11 Patient's oxygen saturation level

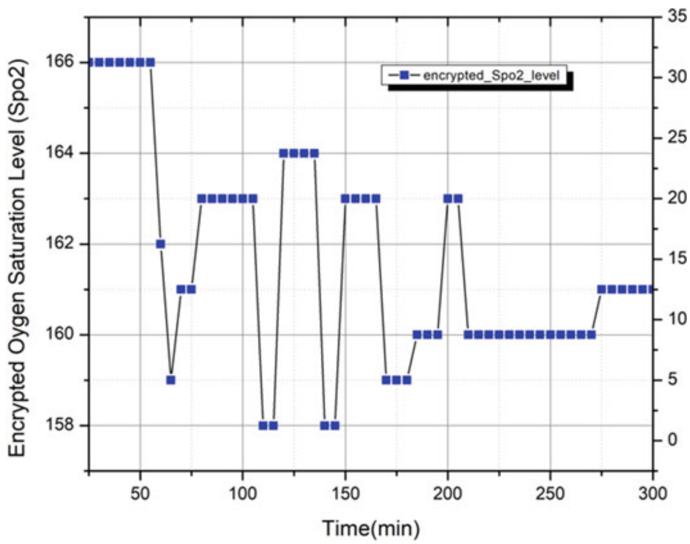


Fig. 12 Encrypted patient's oxygen saturation level

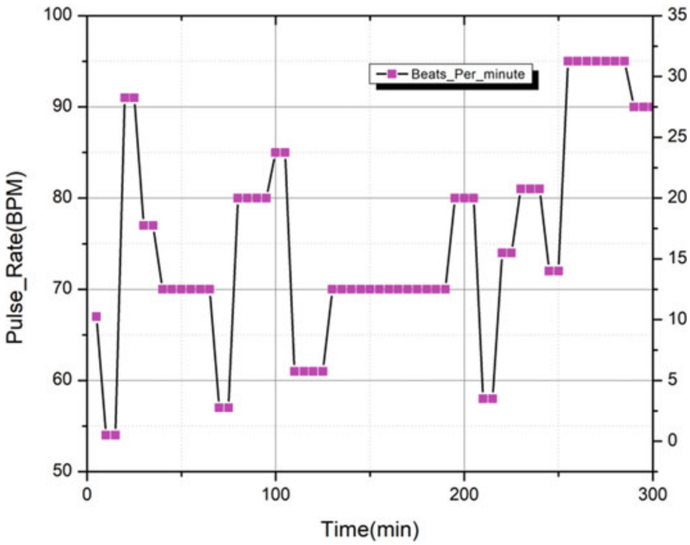


Fig. 13 Patient's heart rate (BPM)

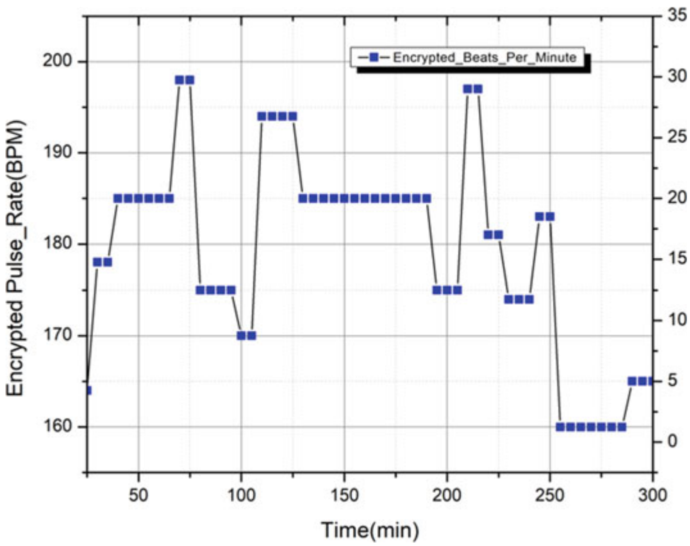


Fig. 14 Encrypted patient's heart rate (BPM)

References

1. Hanoon IK, Aal Nouman MI (2021) Cloud-based COVID-19 patient monitoring using Arduino. In: 2021 3rd East Indonesia conference on computer and information technology (EIConCIT)

April 09–11. ISTTS Surabaya, Indonesia
2. Saha R, Kumar G, Kumar N, Kim T, Devgun T (2021) Internet-of-things framework for oxygen saturation monitoring in Covid-19 environment. IEEE Internet Things J. https://doi.org/10.1109/JIOT.2021.3098158
3. Yoshita Manavi S,Nekkanti V, Choudhary RS (2020) Review on emerging internet of things technologies to fight the COVID-19. In: 2020 Fifth international conference on research in computational intelligence and communication networks (ICRCICN)
4. Utsav A, Abhishek A, Kant K, Badhai RK (2020) Unique identification for monitoring of Covid-19 using the internet of things(IoT). 2020© IEEE
5. Suguna M, Ramalakshmi MG, Cynthia J, Prakash D (2018) A survey on cloud and internet of things based healthcare diagnosis. In: 2018 4th International conference on computing communication and automation (ICCCA)
6. Priamboda R, Kadarina TM (2020) Monitoring self-isolation patient of Covid-19 with internet of things. In: 2020 IEEE international conference on communication, networks and satellite (Comnetsat)
7. Yew HT, Ng MF, Ping SZ, Chung SK, Chekima A, Dargham JA (2020) IoT based real-time remote patient monitoring system. In: 2020 16th IEEE international colloquium on signal processing & its applications (CSPA 2020), 28–29 Feb 2020, Langkawi, Malaysia
8. Saha G, Singh R, Saini S (2019) A survey paper on the impact of "internet of things" in healthcare. In: Proceedings of the third international conference on electronics communication and aerospace technology [ICECA 2019] IEEE Conference Record # 45616; IEEE Xplore ISBN: 978-1-7281-0167-5
9. MacDermott Á, Kendrick P, Idowu I, Ashall M, Shi Q (2019) Securing things in the healthcare internet of things. 978-1-7281-2171-0/19/$31.00 ©2019 IEEE
10. Zhu H, Wu CK, Koo CH, Tsang YT, Liu Y, Chi HR, Tsang K-F (2019) Smart healthcare in the era of internet-of-things. Published by the IEEE Consumer Electronics Society, 2162–2248 2019 IEEE
11. Vibhute M, Deshmukh A, Godse D (2019) Patient monitoring using IoT and medical sensors. In: 2019 5th international conference on computing communication control and automation (ICCUBEA). 978-1-7281-4042-1/19/$31.00 ©2019 IEEE
12. Shukla S, Hassan MF, Jung LT, Awang A (2018) Fuzzy-based fog computing for real-time data transmission in healthcare internet-of-things. In: Second international conference on green computing and internet of things (ICGCIoT), 978-1-5386-5657-0/18/$31.00_c 2018 IEEE
13. Nandyal S, Gada AR (2018) A holistic approach for patient health care monitoring system through IoT. 978-5386-5657-0/10/$31.00© 2018 IEEE
14. Jangra P, Gupta M (2018) A design of real-time multilayered smart healthcare monitoring framework using IoT. In: 2018 International conference on intelligent and advanced system (ICIAS)
15. Gandhi DA, Ghosal M (2018) Intelligent healthcare using IoT: a extensive survey. In: Proceedings of the 2nd international conference on inventive communication and computational technologies (ICICCT 2018) IEEE Xplore Compliant - Part Number: CFP18BAC-ART; ISBN: 978-1-5386-1974-2
16. Salahuddin MA, Al-Fuqaha A, Guizani M, Shuaib K, Sallabi F (2017) Published by the IEEE Computer Society 0018-9162 /17/$33.00 © 2017 IEEE
17. Lavanya S, Lavanya G, Divyabharathi J (2017) Remote prescription and I-Home healthcare based on IoT. In: IEEE International conference on innovations in green energy and healthcare technologies (ICIGEHT'17), Department of BME & EEE, Dr. N.G.P. Institute of Technology
18. Shaikh S, Chitre V (2017) Patient monitoring using IoT. In: Proceedings of the 2nd international conference on communication and electronics systems (ICCES 2017), IEEE Xplore Compliant - Part Number:CFP17AWO-ART, ISBN: 978-1-5090-5013-0
19. Laplante PA, Laplante N (2016) The internet of things in healthcare potential applications and challenges. Published by the IEEE Computer Society 1520-9202/16/$33.00 © 2016 IEEE
20. Gelogo YE, Oh J, Park JW, Kim H-K (2015) Internet of things (IoT) driven u-healthcare system architecture. In: 2015 8th international conference on bio-science and bio-technology, 978-1-4673-9843-5/15 $31.00 © 2015 IEEE. https://doi.org/10.1109/BSBT.2015.17

21. Naveen Kumar K, Raj Kumar GVS, Praveen kumar KT, Chandra Sekhar P (2011) Bitwise operations based encryption and decryption. Int J Comput Sci Eng (IJCSE) 3(1) ISSN : 0975-3397
22. Shaikh S, Chitre V (2017) Healthcare monitoring system using IoT. In: International conference on trends in electronics and informatics ICEI 2017
23. Shaikh Y, Parvati VK, Biradar SR (2018) Survey of smart healthcare systems using internet of things (IoT). In: International conference on communication, computing and internet of things (IC3IoT), 978-1-5386-2459-3/18/$31.00_c 2018 IEEE

TLDC: Tomato Leaf Disease Classification Using Deep Learning and Image Segmentation

Priyanka Sahu, Anuradha Chug, Amit Prakash Singh, and Dinesh Singh

Abstract Deep learning (DL) has made significant progress in identifying and classifying plant diseases. The convolutional neural network (CNN) model was utilized to classify diseased and healthy tomato plant leaves for this study. Seven predominant DL models, namely LeNet 5, AlexNet, VGG19, Inception Net V3, ResNet50, DenseNet 121, and Efficient Net B0 have been used for tomato leaves disease classification. Deep feature extraction and fine-tuning strategies were utilized to adapt these DL models to the specific task of classification. The obtained features using deep feature extraction were then classified by fully connected layers of the CNNs. The experiments were carried out using the image data acquired from the Indian Agricultural Research Institute, India. The dataset consists of diseased and healthy tomato leaf images with a total count of 155 images. Data augmentation was used to increase the dataset size. Furthermore, three segmentation algorithms were also applied to remove the background and highlight the deep features. In this study, a comparison of the above-mentioned CNNs has been carried out to show the accuracy results achieved on the collected dataset. The evaluation results show that deep feature extraction with image segmentation techniques produced better results (up to 100% classification accuracy) than without segmentation. The outcome of this research will have a substantial impact on tomato disease prediction and early prevention.

Keywords Convolutional neural network · Deep learning · Tomato plant · Classification · Image pre-processing · Disease inoculation

P. Sahu (✉) · A. Chug · A. P. Singh
University School of Information, Communication, and Technology, Guru Gobind Singh Indraprastha University, New Delhi, India
e-mail: er.priyankasahu40@gmail.com

A. Chug
e-mail: anuradha@ipu.ac.in

A. P. Singh
e-mail: amit@ipu.ac.in

D. Singh
Division of Plant Pathology, Indian Agricultural Research Institute, New Delhi, India
e-mail: dinesh_iari@rediffmail.com

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_35

1 Introduction

The advancement of computer vision models has accelerated efforts to create automated systems for detecting plant diseases based on visible lesions on leaves. These systems attempt to make farmer participation as simple as feasible while also ensuring that the detection system is as reliable as possible. Prior to the widespread availability of deep learning (DL) models, researchers primarily relied on image processing or feature extraction to build disease detection algorithms, with inconsistent results [1–5].

The most difficult aspect of these manually handcrafted techniques is to define the symptoms to the computer for the disease recognition, which has been overcome by employing DL, especially convolutional neural networks (CNNs), which allow models to learn features rather than having to explain them. In recent years, a lot of research has successfully achieved varied levels of accuracy on laboratory/real-field images using DL models [6–13]. These deployed classifiers attain highly accurate results when evaluated on data that is identical to that used during the training process. However, when tested on a completely different dataset, their performance falls drastically. In a study [14], the authors have been identified a set of factors that influence the success of DL techniques implemented for plant leaf disease classification and deduced that, despite the high accuracy rate of developed models, including their own, there are a plethora of factors that show why it is still far from being a standardized tool that can be used in real-world scenarios.

For the relatively limited size of the datasets, training the DL models from scratch degrades the accuracy. In order to enhance the same, transfer learning comes which overlays the outcome of a DL model to a new dataset has become a prominent choice for researchers. [15–17] are a few examples of using transfer learning techniques and many of them have used the openly accessible image datasets such as PlantVillage [18]. The PlantVillage dataset, which comprises images with low variance and uniform backgrounds, has been used to train and evaluate many of the DL models as reported in the literature in order to detect the disease present in the leaves of the plants. In [19], authors have focused on the performance of DL models trained on individual lesions and spots, applying image segmentation and augmentation to expand the dataset from a limited number of images.

This study follows a similar concept: image files are segmented to get the most information about the disease symptoms. By comparing and quantifying the performance of seven models, it is demonstrated that we may train with more meaningful data and achieve excellent results even in real-world scenarios. The main contributions of the study are:

- A real-field image dataset with 155 tomato leaf images has been acquired from the Indian Agricultural Research Institute, India.
- A comparison of seven state-of-the-art DL models has been performed for tomato leaf disease classification.
- K-means, Otsu, and watershed algorithms have been deployed to enhance the features of leaves.

The motive of the study is to deploy the DL models in order to predict the tomato crop leaf diseases at an early stage of infection that would help the farmers to diagnose the crops with the appropriate action on time. The paper comprises the following sections: Sect. 2 describes the materials and methods, as well as the research methodology and other implementation specifics; Sect. 3 covers the implementation results and analyses the outcomes; and finally, Sect. 4 summarizes the research findings and suggests some future research paths.

2 Materials and Methods

2.1 Dataset Collection

Authors have a target to build their own image dataset for tomato crop leaves. In order to achieve this, a collaboration was done with the Indian Agricultural Research Institute (IARI) in New Delhi, India. At IARI, forty tomato pots were planted under a greenhouse as shown in Fig. 1. All planted pots have been inoculated artificially with a culture of pathogen: Alternaria solani, which causes early blight disease in tomato crops. A digital SLR camera, Canon EOS 1500D 24.1 (EF S18-55), with II lens was used to capture the diseases, and healthy images of the leaves were taken from the tomato pots. Initially, images were taken in RAW format with an aspect ratio of 16:9, and then accordingly converted in the JPEG format. All the acquired images were set to a dimension of 256 * 256.

The leaves were photographed from a consistent distance on a flat uniform background. The camera lens was placed perpendicular to the blade surface, and the focal length was adjusted to a fixed point. The leaves were chosen based on three principles: (1) tomato leaves are flat and easy to photograph; (2) the surfaces are clean; and (3) the periodic disease features are easily identifiable.

Fig. 1 Tomato plantation pots (healthy and inoculated with Alternaria solani pathogen) in a controlled environment of greenhouse at IAR

Table 1 Dataset summary

Class name	# Original images	# Augmented images
Early blight	79	678
Healthy	76	681
Total	155	1359

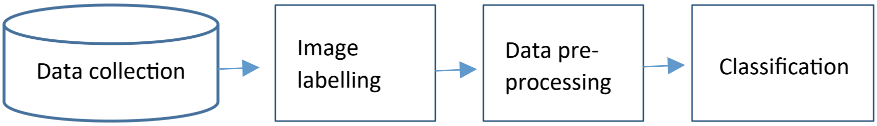


Fig. 2 Generalized approach of disease classification

2.2 *Labelling and Image Pre-processing*

The original images have been cropped and scaled to 256 * 256 dimensions from the raw image format after data collection. Subsequently, data augmentation was used to add diversity to the dataset in order to better mimic the real-world circumstance. Rotation, skew, brightness, height shift, width sift, zoom, mirror, shear, and horizontal flips were some of the operations used to augment the images of the collected dataset, thus, help to enhance the limit of the dataset size. Furthermore, image segmentation was also performed to highlight the features and to segment the desired portions of an image. Image segmentation also helps to remove the unwanted background and other information. In this study, three image segmentation algorithms have been deployed, namely the Otsu segmentation algorithm, K-means algorithm, and watershed algorithm. Hence, four datasets were formed, three with segmentation algorithms and one without segmentation. Image pre-processing was performed by utilizing the image processing algorithms in Google Colab, Python 3. Table 1 summarizes the total of original and augmented images. Moreover, the disease classification methodology of a generalized DL algorithm can be represented as shown in Fig. 2.

3 Experimental Setup and Result Discussions

In the study, binary classification (two-class) was performed for healthy and diseased tomato leaves. For disease classification and prediction, seven pre-trained models were deployed on the dataset using fine-tuning of DL models. All the experiments were carried with 70, 15, and 15% for training, testing, and validation dataset. The implementation was performed using the Python Google Colab notebook for 20 epochs, and accuracy, loss, precision, recall, and F1-score were used as the performance measurement criteria. Table 2 shows the implementation details of the deployed CNN models. Furthermore, SGD was used as an optimization technique

Table 2 Implementation details of the deployed CNNs

DL model	Input dimensions	Parameters	Depth	Size (MB)	Model execution using pre-trained weights: time (ms) per inference step (GPU)
LeNet5	32 * 32	60 thousand	5	42	3.23
AlexNet	227 * 227	60 million	8	227	3.15
VGG19	224 * 224	138 million	19	549	4.38
Inception V3 Net	299 * 299	24 million	48	92	6.86
ResNet 50	224 * 224	26 million	50	98	4.55
DenseNet 121	224 * 224	20 million	121	33	5.38
Efficient Net B0	224 * 224	5 million	18	29	4.91

and the batch value was set at 16. In addition, an initial learning rate of 0.001 was utilized along with a momentum value of 0.9. All the implementations have been carried for 20 epochs.

Tables 3 and 4 illustrate the performance metrics for each implemented model, whereas Figs. 3 and 4 show the comparison charts of disease classification that have been emerged.

As can be seen, Efficient Net B0 performed best in two classes without segmentation, with 98.42% accuracy and minimal validation loss of 0.012. Whereas, LeNet gave performed worst with 84.83% accuracy and also shows the highest loss of 0.638. In addition, all implementations increased accuracy metrics after applying segmentation techniques with Efficient Net B0 providing 100% accuracy and the least validation loss. After deploying all the CNNs, a mixed set of results were observed**.**

Table 5 shows the comparison of the proposed approach with some state-of-the-art methods.

Table 3 Performance comparison of implemented CNNs without segmentation and segmented using Otsu algorithm

DL model	Before segmentation			Segmentation using Otsu algorithm		
	Accuracy (%)	Loss	F1-score	Accuracy (%)	Loss	F1-score
LeNet 5	84.83	0.638	0.83	85.48	0.526	0.85
AlexNet	90.51	0.482	0.90	92.04	0.264	0.92
VGG19	95.21	0.136	0.94	97.86	0.104	0.97
Inception Net V3	98.07	0.092	0.98	100	0.041	1
ResNet 50	97.45	0.071	0.97	99.98	0.042	0.99
DenseNet 121	96.88	0.032	0.97	98.66	0.016	0.98
Efficient Net B0	98.42	0.012	0.98	100	0.005	1

Table 4 Performance comparison of implemented CNNs after segmenting with K-means and watershed algorithm

DL model	Segmentation using K-means algorithm			Segmentation using watershed algorithm		
	Accuracy	Loss	F1-score	Accuracy	Loss	F1-score
LeNet 5	84.11	0.426	0.84	82.48	0.796	0.82
AlexNet	90.55	0.284	0.91	89.14	0.364	0.91
VGG19	93.40	0.204	0.93	95.86	0.194	0.95
Inception Net V3	99.78	0.048	0.99	100	0.025	1
ResNet 50	99.99	0.062	0.99	100	0.024	1
DenseNet 121	100	0.009	1	99.26	0.019	0.99
Efficient Net B0	100	0.002	1	100	0.001	1

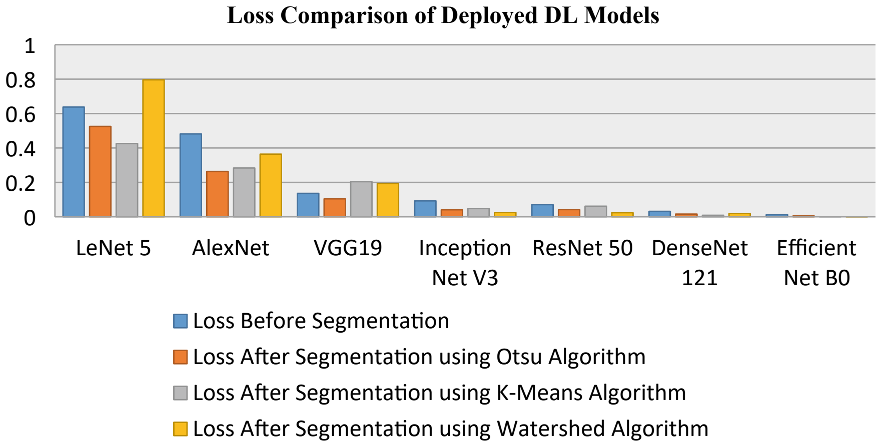


Fig. 3 Loss comparison of deployed CNNs

4 Conclusion

In this study, the performance of seven DL models, namely LeNet 5, AlexNet, VGG19, Inception Net V3, ResNet50, DenseNet 121, and Efficient Net B0 have been implemented and compared for tomato leaves image classification. Deep features have been extracted using the above-mentioned CNNs, and classification was done using the fully connected layers of the deep networks. A dataset of 155 original acquired images with early blight diseased and healthy tomato leaves was collected and pre-processed; thereafter, image augmentation was used to enhance the size of the acquired dataset. For the task of disease detection and classification, accuracy varies in the range of 84–100% after deploying the CNNs. Furthermore, three image segmentation techniques, namely Otsu, k-means, and watershed algorithm were also

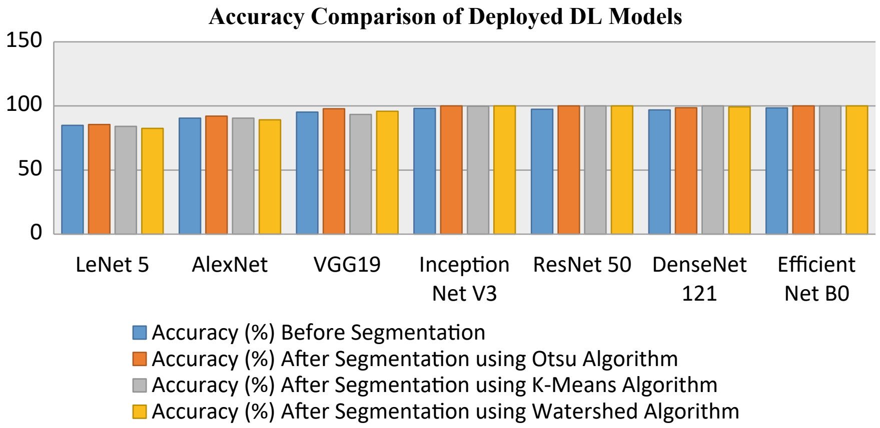


Fig. 4 Accuracy comparison of deployed CNNs

Table 5 Comparison with a few benchmark frameworks

DL model	Images	Dataset	Classes	Results	References
AlexNet	54,306	PlantVillage	38	Classification accuracy, 93.88%	[6]
Faster R-CNN, R-CNN, SSD, VGG, ResNet	5000	Own	9	mean AP, 71.1–85.98%	[7]
2D-CNN bidirectional gated recurrent unit neural network	90	Own	3	Accuracy, 74.3%	[8]
GMDH-logistic model	80	Own	4	Average recall, 86.67%	[16]
Seven CNNs	155	Own	2	Classification accuracy, 84–100%	Proposed

applied in order to enhance the features present in the images. It was observed that accuracy goes to 100% after deploying the segmentation algorithms on this limited dataset. In future, the authors have the target to collect images of some of the other diseases also to add to the collected dataset. The primary goal of the future study will be to create a smart mobile device app that can identify various plant infections. Individuals with little to no understanding of the plants they are growing could profit them greatly from this programme, which will give automatic plant disease diagnosis with image analysis.

Acknowledgements Authors are thankful to the Department of Science & Technology, Government of India, Delhi, for funding a project on “Application of IoT in Agriculture Sector” through the ICPS division. This work is a part of the project work.

References

1. Kruse OMO, Prats-Montalbán JM, Indahl UG, Kvaal K, Ferrer A, Futsaether CM (2014) Pixel classification methods for identifying and quantifying leaf surface injury from digital images. Comput Electron Agric 1(108):155–165
2. Clément A, Verfaille T, Lormel C, Jaloux B (2015) A new colour vision system to quantify automatically foliar discolouration caused by insect pests feeding on leaf cells. Biosyst Eng 1(133):128–140
3. Barbedo JGA, Koenigkan LV, Santos TT (2016) Identifying multiple plant diseases using digital image processing. Biosyst Eng 1(147):104–116
4. Radovanović D, Đukanovic S (2020) Image-based plant disease detection: a comparison of deep learning and classical machine learning algorithms. In: 2020 24th International conference on information technology (IT), pp 1–4
5. Sujatha R, Chatterjee JM, Jhanjhi NZ, Brohi SN (2021) Performance of deep learning vs machine learning in plant leaf disease detection. Microprocess Microsyst 1(80):103615
6. Mohanty SP, Hughes DP, Salathé M (2016) Using deep learning for image-based plant disease detection. Front Plant Sci 7:1419
7. Fuentes A, Yoon S, Kim SC, Park DS (2017) A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. Sensors 17(9):2022
8. Jin X, Jie L, Wang S, Qi HJ, Li SW (2018) Classifying wheat hyperspectral pixels of healthy heads and Fusarium head blight disease using a deep neural network in the wild field. Remote Sens 10(3):395
9. Saleem MH, Potgieter J, Arif KM (2019) Plant disease detection and classification by deep learning. Plants 8(11):468
10. Johannes A, Picon A, Alvarez-Gila A, Echazarra J, Rodriguez-Vaamonde S, Navajas AD et al (2017) Automatic plant disease diagnosis using mobile capture devices, applied on a wheat use case. Comput Electron Agric 138:200–209
11. Kamilaris A, Prenafeta-Boldú FX (2018) Deep learning in agriculture: a survey. Comput Electron Agric 147:70–90
12. Sravan V, Swaraj K, Meenakshi K, Kora P (2021) A deep learning based crop disease classification using transfer learning. Mater Today Proc
13. Sharma P, Berwal YPS, Ghai W (2020) Performance analysis of deep learning CNN models for disease detection in plants using image segmentation. Inf Process Agric 7(4):566–574
14. Barbedo JGA (2018) Factors influencing the use of deep learning for plant disease recognition. Biosyst Eng 172:84–91
15. Too EC, Yujian L, Njuki S, Yingchun L (2019) A comparative study of fine-tuning deep learning models for plant disease identification. Comput Electron Agric 161:272–279
16. Chen J, Yin H, Zhang D (2020) A self-adaptive classification method for plant disease detection using GMDH-Logistic model. Sustain Comput Informatics Syst 28:100415
17. Chen J, Zhang D, Nanehkaran YA (2020) Identifying plant diseases using deep transfer learning and enhanced lightweight network. Multimed Tools Appl 79(41):31497–31515
18. Hughes D, Salathé M et al (2015) An open access repository of images on plant health to enable the development of mobile disease diagnostics. arXiv Prepr arXiv: 151108060
19. Barbedo JGA (2019) Plant disease identification from individual lesions and spots using deep learning. Biosyst Eng 180:96–107

A Novel Deep Supervised Contour Fractal Dimension Analysis Model

Abirami Balasubramanian and Krishnaveni Krishnasamy

Abstract A novel palmprint recognition system (PRS) using deep supervised learning (DSL) classifier is proposed in this research work. To divulge the novelty, a deep supervised contour fractal dimension analysis model for palmprint recognition (DCFPR) is put forward. That has a novel region-based contour fractal dimension (RCFD) feature extraction approach and a deep supervised Learning (DSL) classifier approach for acquiring the higher recognition and identification accuracy rate. To accomplish the RCFD approach, traced all the edges/contours of 2D palmprint region of interest (2D-PROI) image using Canny edge detection algorithm and then split into several regions. At each region, fractal dimension (FD) and the slope value (S) are computed in an idiosyncratic manner using the box-counting procedure and then accumulate all FDs and Ss of all regions to create a distinctive feature vector. Classify this feature vector using deep supervised learning (DSL) classifier approach to authenticate the genuine person of the taken palmprint at a higher accuracy rate. In this research, the multi-spectral 2D-PROI image database derived from PolyU, Hong Kong Polytechnic University, Hong Kong. The proposed model has been examined and evaluated with various metrics and found with 98% of authentication accuracy.

Keywords Palmprint recognition system · Deep supervised learning classifier · Region-based contour fractal dimension · Canny's edge detection algorithm · Box-counting · Fractal dimension

1 Introduction

Biometric imparts a secured technique of substantiating identification [1]. Biometric authentication and identification systems (BAIS) are typically necessary to protect the emerging digital aspects of the IoT world. Accordingly, the benefaction of this research is to make adequate preparation for furnishing an optimal remedy for the

A. Balasubramanian (✉) · K. Krishnasamy
Sri S. Ramasamy Naidu Memorial College, Sattur, India
e-mail: abirami@vhnsnc.edu.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_36

following: (a) to overcome the demands of users who are wanted to secure their e-information and worthy assets at low cost in a more truthful and opportune manner, (b) to provide the finest security to our society from the unauthorized intruders' hacking, (c) to provide a perfect solution to avert illicit access to digital information.

BAIS is a programmed real-time application of human identification which can be possible by acquiring, determining, and scrutinizing the digital data of human corporeal and logical characteristics [2]. Researchers are exploring several biometric traits to earn the peculiar BAIS. However, many researchers are impleading their exploration into PRS due to its singularity, stability, and high-performance characteristics [3]. The large space of the palmprint area includes principle lines and wrinkles which provide lots of unique information compared to the fingerprint area. Human palmprint principle lines and wrinkles can be treated as edges or contours for receiving the feature information. And those edges are not identical to each palm in a human hand. The Canny edge detection algorithm classifies the perfect set of edges of various sizes in an entire image [4].

Humeau-Heurtier [5] proposed segmentation-based fractal analysis to apply fractal model at low-computation time compared to other techniques. Fractal dimension (FD) approach is considered as a widely applied descriptor, especially for analyzing the texture representation, in several fields like the signature recognition, palmprint recognition, written identification, etc. [6, 7]. Shi et al. [8] stated that the fractal-based feature extraction technique has been proved as the agreeable approach in a computer-aided diagnosis system with a huge dataset. In 1983, Mandelbrot introduced fractal geometry to define fractals as the sets of self-identical. Fractal dimension is a ratio that describes the asymmetry and the difficulty of the stochastic models, indicating the pattern changes feature at the various scale [9]. The box-counting algorithm is the most progressive and straightforward technique to measure the fractal dimension of the image [1, 10]. It works well for both linear and nonlinear fractal images and derives the deformity patterns on the surface of the images [11, 12].

A set of feature vectors is created and stored for identification and recognition processes by the DSL classifier algorithm which is the sub-area of artificial intelligence (AI) technology. Ata et al. [13] proved that feed-forward neural network with Back-Propagation achieved the higher authentication precision of 99.99% compared to other machine learning techniques used in this paper. Dandan et al. [14] showed that the deep learning method has yielded higher recognition rate for palmprint images. Machine learning algorithms forecast the outcome in higher accuracy using anterior knowledge of content. Nevertheless, massive datasets are not learned thoroughly in the machine learning algorithms that cause the lack of limpidity, interpretability in the decision-making. To overcome these drawbacks, researchers have extended their view of perception toward the next level that is the deep learning classifier approach [15] described in Sect. 2. Deep learning on massive datasets is being most potent in the medical and biometric field, and its enhanced techniques train the datasets faster and accurately [16].

Consequently, this paper employs an innovative RCFD feature extraction approach to brand the simple and fast computation. It computes the fractal dimension and slope values from the contours of the 2D-PROI images in an idiosyncratic manner to discrete the distinct features. That is explained in Sect. 3. A suitable DSL classifier algorithm and its configuration are designed for setting the higher rate of recognition, and verification precision rate at a low time, which is explained in Sect. 3.

2 General Description of Deep Supervised Learning (DSL) Classifier

A deep neural network (DNN) is an artificial neural network (ANN). It learns large input for predicting the exact outcome at a low error cost [17]. The general DNN architecture and the implementation of DSL classifier algorithm are discussed below.

2.1 DNN Architecture

DNN architecture includes an input layer (x^i) where i refers to the value of training template i-vector values from 1 to N (i.e., N is the maximum number of input data), an output layer (y^j) where j refers to the target j-vector values from 1 to M (i.e., M is the maximum number of output data), and the multiple hidden layers hk^p where k refers to the number of hidden layers and p refers to the number of hidden perceptrons. The values of k and p are regulated progressively until least error rate is reached at a minimum training time. The learning rate value α and the epoch value $E\gamma$ are used to decrease the gradients and to know the number of iterations needed to attain the least mean square error through the whole dataset where γ refers to the maximum number of iterations. Next, the weight matrices are assigned to all the layers of DNN that are labeled as $Wm1$, $Wm2$, and $Wm(k-1)$ where m refers to the number of weight matrices and k refers to the number of hidden layers in the network. The basic DNN architecture is shown in Fig. 1.

2.2 Implementation of DSL Algorithm

Implementation of the DSL algorithm is done in two aspects: (1) training aspect and (2) testing aspect. At the training aspect, the supervised learning algorithm is applied on DNN where two processing stages are performed in each epoch. In the first stage, the signal passes from the input perceptrons (x^i) to the first hidden layer h^p perceptrons along with the output using Eqs. (1) and (2). Similarly, the output

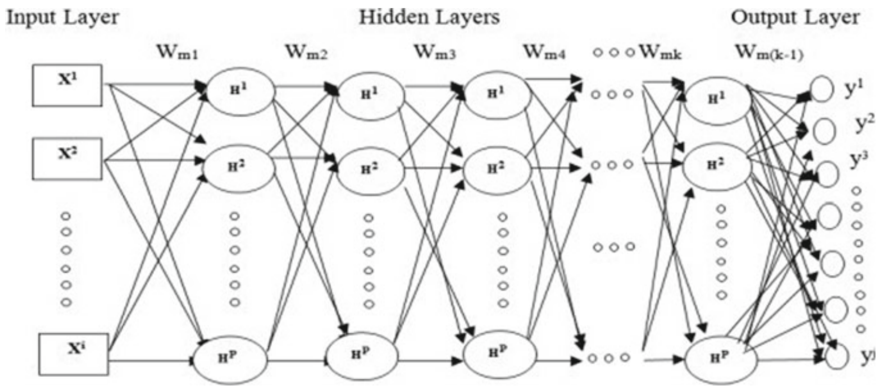


Fig. 1 Basic architecture of DNN

of all hidden layers is gained till the output perceptrons are reached. In the output layer, output is calculated using (3) and to get the final output p^j. Each hidden unit determines its error by using the output unit's error.

$$h_j^p = w_m \times x^i|i = 1, \ldots, N||m = 1, \ldots., (k-1)||j = 1, \ldots, k| \tag{1}$$

$$h_j^p = \beta\left(h_j^p\right) \quad |j = 1, \ldots, k| \tag{2}$$

$$p^j = \beta\left(w_{m(k-1)} \times y^j\right)|m = 1, \ldots., (k-1)||j = 1, \ldots, M| \tag{3}$$

The dissimilarity between the exact output y^j and the final output of the network p^j is computed as an error Δ^j using (4).

$$\Delta^j = y^j - p^j|j = 1, \ldots, M| \tag{4}$$

In the second stage, the weights of all layers are updated using (5) in the backward direction along with the reverse processing procedure of the first stage to decrease the gradient.

$$w_m^{'} = \alpha \times \Delta^j \times y^j|m = k-1, k-2, \ldots, 1||j = 1, \ldots, M| \tag{5}$$

Similarly, the first and second stage processes are repeated iteratively with the updated weights until the least mean square error (MSE) rate between the prediction and true class values is achieved. Thus, the DSL classifier algorithm is tuned to learn the input and map the output exactly for the recognition process.

And at the testing aspect, tuned DSL classifier with the obtained weight values is used to determine the verification accuracy.

3 Proposed Methodology

The proposed deep supervised contour fractal dimension analysis model for palmprint recognition (DCFPR) has been performed in four levels: (1) data acquiring level, (2) preprocessing level, (3) feature extraction level, and (4) classification or matching level [1]. The recognition and verification processes of DCFPR are depicted in Figs. 2 and 3.

3.1 Data Acquiring

This research explores the 2D-PROI images of a multi-spectral 2D-ROI palmprint image database at the biometric research center (UGC/CRC) in POLYU, Hong Kong.

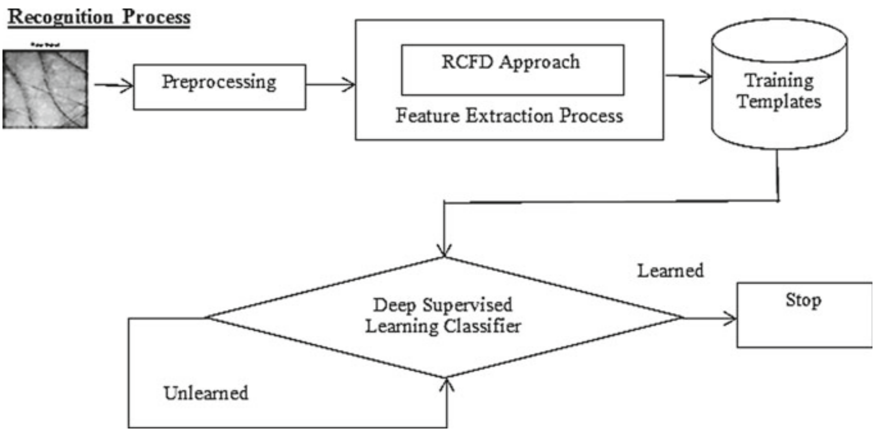


Fig. 2 Block diagram of DCFPR recognition process

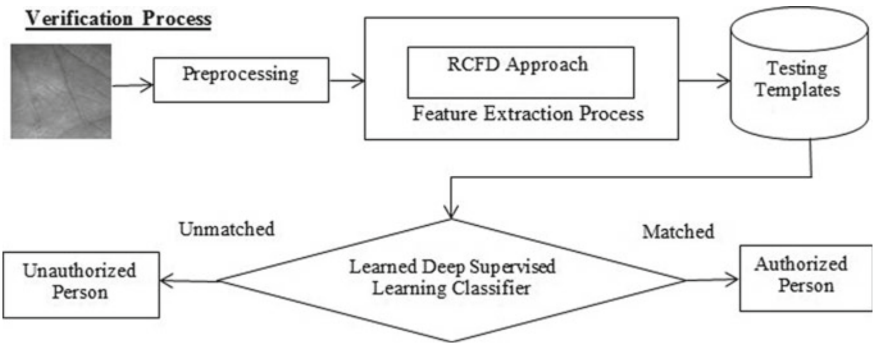


Fig. 3 Block diagram of DCFPR verification process

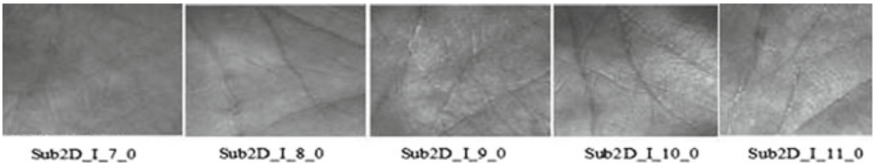


Fig. 4 POLYU 2D-PROI bitmap images

That is comprised of 8000 segmented and normalized BMP image files of 400 volunteers' left and right hands' palms, store in two separate sections. Each section has ten images of each palm and some of these sample 2D-PROI images are shown in Fig. 4.

3.2 Preprocessing

The input image is molded to the well-defined quality image by doing the following processes: (1) convert the input image into gray-scale image, (2) remove the unfavorable data in the image, and (3) increase the spatial intensity level of the image. By preprocessing, the tiny creases, lines, and ridges of palmprint are spotted out more visibly as shown in Fig. 5. Then Canny edge detection algorithm is applied for tracing the lines, ridges, and curves, and the resultant image is represented as contour palmprint image (CI), shown in Fig. 6.

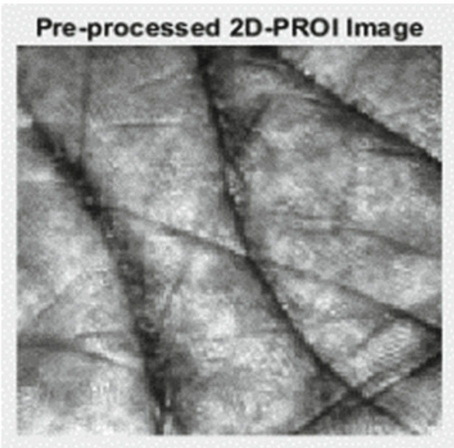


Fig. 5 Preprocessed image of 2D-PROI image

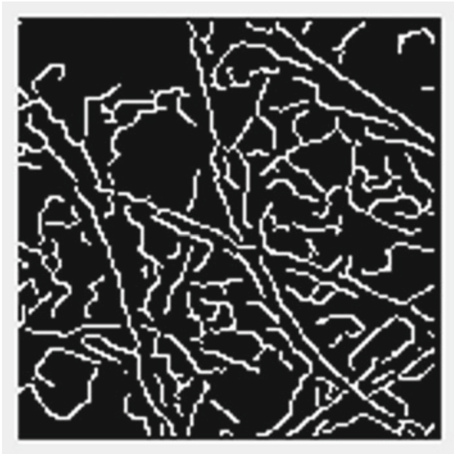


Fig. 6 Canny edge detection on the preprocessed image

3.3 *Feature Extraction Level*

A set of discrete model-based texture features of an image can be effectively excerpted by exploiting the fractal dimension (FD) approach [7]. This paper is explored and furnished the RCFD feature extraction approach to appraise the FD values of an input image in an idiosyncratic manner. The first step of the implementing the RCFD approach is to split the contour palmprint image (CI) into four quarters (one-fourth) of nonoverlapping regions with the equal size $\Lambda \times \Lambda$ where Λ refers to the dimension of each region [18]. Each region is labeled as (CIr) where r refers to the number of regions, i.e., r =1, 2, 3, and 4, which is shown in Fig. 7. Next, the fractal dimension FDr and slope Sr values for all regions (CIr) are found by exploiting box-Counting algorithm in a novelty manner. Finally, the representative FD and its S values are determined as discussed in the below subheading and Fig. 8 depicts the graphical representation of RCFD approach.

3.3.1 RCFD Approach

The procedure adapted in RCFD approach is given below.

a. The fractal dimension (FDr) and its slope (Sr) for each region of CI are calculated using the box-counting algorithm in a specific manner.
b. Determine the mean value of all edges' length ($\mu\Phi$) in all square boxes $\Phi \times \Phi$ of set ε, where $\Phi = \Lambda/\varepsilon$ at various box scale intervals ε, ($\varepsilon = 1, \ldots, \Psi$) where ψ refers to the maximum box scale interval value (i.e., log2(Λ)/2) using (6)

$$\mu_{\Phi,\varepsilon} = \sum_{\varepsilon=1}^{\Psi} \frac{\sum_{\Phi=1}^{\frac{\lambda}{\mathrm{E}}} L_\Phi}{E_\Phi} \bigg| \Phi = 1, \ldots, \frac{\lambda}{\mathrm{E}} \bigg| |\varepsilon = 1, \ldots, \Psi| \tag{6}$$

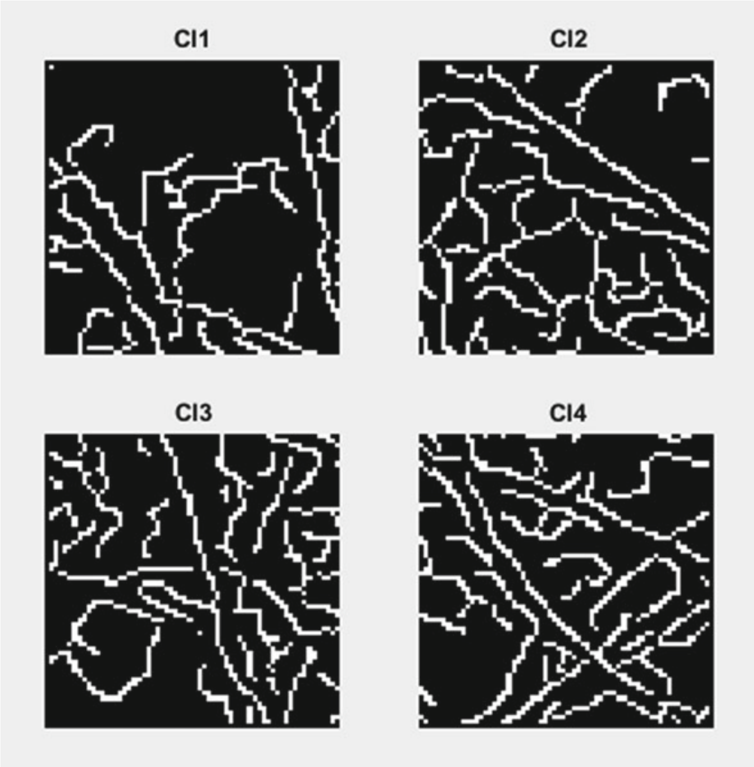


Fig. 7 Equal quarter size nonoverlapping regions of CI (labeled as CI1, CI2, CI3, and CI4.)

where $L\Phi$ and $E\Phi$ refer to the edge length and the total number of edges within square boxes $\Phi \times \Phi$ of set ε.

c. Find the probability mass function (PMF_ε) of each set ε using (7).

$$\text{PMF}_\varepsilon = \sum_{\varepsilon=1}^{\Psi} \frac{\sum \mu_{\Phi,\varepsilon}}{N_\varepsilon} \left| \Phi = 1, \ldots, \frac{\lambda}{\text{E}} \right| |\varepsilon = 1, \ldots, \Psi| \tag{7}$$

where $N\varepsilon$ refers to the total number of boxes in the set ε.

d. Estimate the (FDr) and its slope value (Sr) of each set ε for all regions of CI using (8)

$$\text{FD}_{r,\varepsilon} = \sum_{\varepsilon=1}^{\Psi} \frac{\log(\text{PMF}_\varepsilon)}{\log\left(\frac{1}{2^\varepsilon}\right)} |r = 1, 2, 3, \text{ and } 4||\varepsilon = 1, \ldots, \Psi| \tag{8}$$

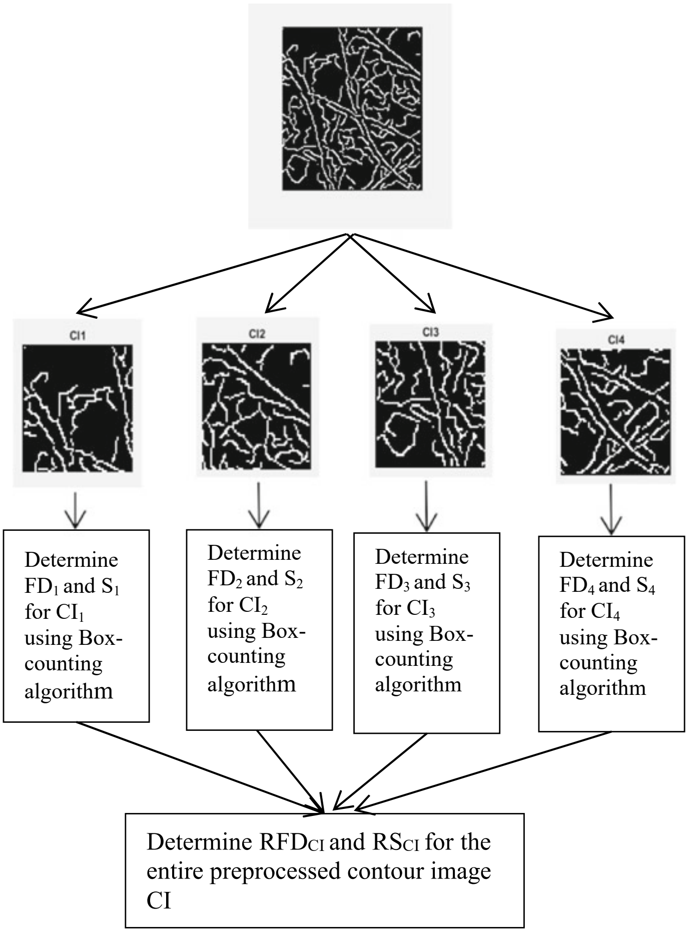


Fig. 8 Graphical representation of RCFD approach

And this log transformation yields a straight line with slope Sr, ε for each set ε to know more even spatial distribution and degree of high intensity values in an image.

e. Finally, calculate the representative fractal dimension (RFDCI) and slope (RSCI) values of CI by using (9) and (10).

$$\mathrm{RFD}_{CI} = \sum_{r=1}^{4} \frac{\sum_{\varepsilon=1}^{\Psi} \mathrm{FD}_{r,\varepsilon}}{\sum \sum_{\varepsilon=1}^{\Psi} \mathrm{FD}_{r,\varepsilon}} |r = 1, 2, 3, 4||\varepsilon = 1, \ldots, \Psi| \quad (9)$$

$$\mathrm{RS}_{\mathrm{CI}} = \sum_{r=1}^{4} \frac{\sum_{\varepsilon=1}^{\Psi} S_{r,\varepsilon}}{\sum \sum_{\varepsilon=1}^{\Psi} S_{r,\varepsilon}} |r = 1, 2, 3, 4||\varepsilon = 1, \ldots, \Psi| \quad (10)$$

Thus, RFDCI and SCI values are made out and stored as the training and testing template datasets using RCFD approach.

4 Classification or Matching Level

According to the general description of the DSL classifier algorithm described in Sect. 2, the DSL classifier is developed and tuned with the suitable activation functions, hidden layers, and its weight values to absorb the training template accurately in the recognition process. This tuned DSL classifier uses ReLU activation function to calculate the output of the hidden layers for avoiding the problem of vanishing gradient, and rise up the translational invariance to reinforce the output [16] and the sigmoid activation function is used to find the output perceptrons' values in the output layer for setting the probability of the output data range between 0 and 1.

5 Experimental Analysis

This research is appropriated 400 2D-PROI images of 20 volunteers in POLYU database, on that 80% of training image samples and 20% of testing image samples are deployed for the palmprint identification process. In the testing phase, initially, 100 image samples have been taken and further increased in steps of 100 for each testing. Primarily, training and testing templates are generated using RCFD feature extraction approach. Figure 9 shows the resultant measurement of fractal dimension FD1 and its slope $S1$ values of CI1 of the test image. Likewise, FD and its S values

ε	2^ε	$\Phi \times \Phi$	N_ε	μ_Φ	p_ε	$\mathrm{Log}(1/2^\varepsilon)$	$\mathrm{Log}(p_\varepsilon)$	FD_1	S_1
1	2	32 × 32	4	18.7750	9.3875	-0.6931	2.2394	-3.2310	1.9082
2	4	16 × 16	16	11.5490	2.8872	-1.3863	1.0603	-0.7648	
3	8	8 × 8	64	5.3307	0.6663	-2.0794	-0.4060	0.1952	

Fig. 9 Fractal dimension (FD1) and its slope ($S1$) value of CI1 of a test image

for all regions of CI of the test image are estimated using (8). Finally, using (9) and (10) RFDCI values 1.5920, 0.3878, and −0.0734 and SCI value 0.9059 is evaluated which revealed the degree level of gray intensity values in the test image.

In training and testing phases, DNN consists of five layers, where the first layer is the input layer that contains input nodes of the RCFD approach's feature *i*-vector consisting of three fractal dimension values and the corresponding slope values, the last layer is the output layer that carries $\lfloor \log 2n \rfloor + 1$ output nodes where n refers the number of bits required to denote the total number of training dataset, and three hidden layers are proposed in DCFPR system. As illustrated in Sect. 2, DNN is trained with DSL classifier algorithm. The completion of the training process is assessed through the mean square error (MSE) values. It is assumed that the DSL classifier algorithm trains the dataset very well if the MSE value is very least. The metric graph of the training process is shown in Fig. 10 in that MSE values are approaching to the least MSE value at the epoch 9000. It can be effectuated by the salient parameters' values used in the DSL classifier algorithm, which is reported in Table 1.

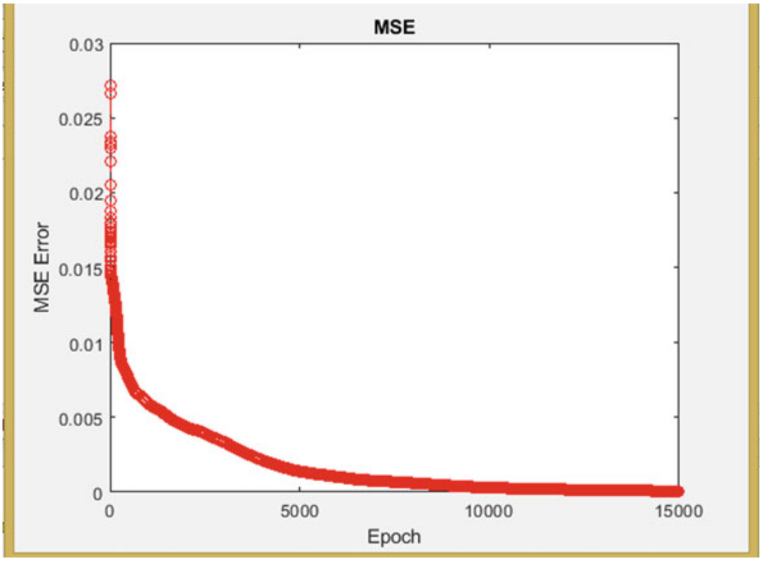


Fig. 10 Metric graph of DRCPR system training algorithm

Table 1 Salient parameters' values used in the training phase of DRCPR system

Number of testing samples	Number of volunteers' palms used	Number of input nodes	Number of output nodes	Epoch (γ)	Number of hidden layers	Number of hidden nodes or perceptrons	Learning rate
400	20	4	10	9000	3	33	0.00001

The DCFPR system classifier performance is scrutinized through confusion matrix parameters such as Accuracy or Correct Positive Rate (CPR), True Positive Rate (TPR), False Positive Rate (FPR), Precision, and Specificity. In particular, the predicted values (True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN)) are computed to measure the parameter values of the confusion matrix. Those values are obtained at the testing phase, where a set of the testing template compared with a set of the trained template and counted that how many testing datasets are matched correctly or not. Those predicted values are substituted in the confusion matrix parameters using (11), (12), (13), (14), and (15).

(1) True Positive Rate (TPR)

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{11}$$

(2) False Positive Rate (FPR)

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} \tag{12}$$

(3) The Accuracy or Correct Positive Rate (CPR)

$$\mathrm{CPR} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{13}$$

(4) Precision

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{14}$$

(5) Specificity

$$\mathrm{Specificity} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \tag{15}$$

Figure 11 notifies the increasing value of True Positive (TP) in every increasing testing dataset. Table 2 and Fig. 12 signify the parameters of the confusion matrix that shows the increasing range of testing image sample size (100, 200, 300, and 400), and its corresponding increasing value of TPR (0.90721, 0.97326, 0.993031,

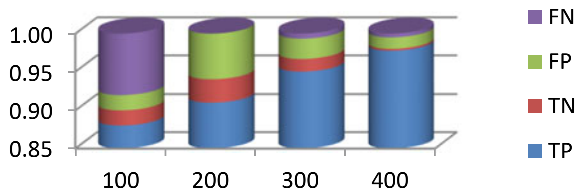


Fig. 11 Predicted values of the confusion matrix for various ranges of testing template

Table 2 Predicted values and confusion matrix parameters

Number of testing samples	TP	T N	F P	F N	Precision (%)	Specificity (%)	TPR (%)	FPR (%)	CPR (%)
100	88	2	1	9	98.8764	66.6666	90.7215	33.3333	90
200	182	6	7	5	96.2962	46.1538	97.3262	53.8461	94
300	285	5	8	2	97.2696	38.4615	99.3031	61.5384	96.666
400	391	1	5	2	98.7373	16.6666	99.2385	83.3333	98

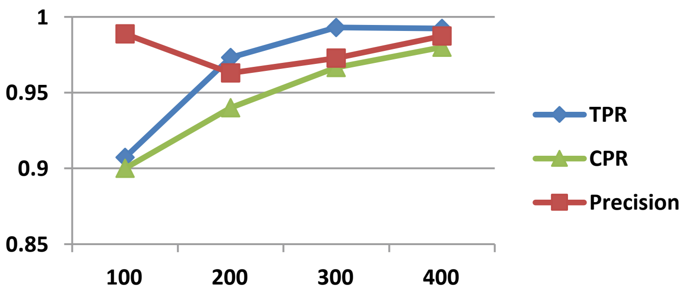


Fig. 12 Confusion matrix parameter values for various ranges of testing template

and 0.992385) and CPR values (0.90, 0.94, 0.9666 and 0.98). It implies that the increasing values of TPR and CPR are directly proportional to the increasing range of image samples [19]. That is demonstrated the proposed DCFPR system performs large datasets accurately and efficiently. As shown in Table 3 and Fig. 13, the proposed

Table 3 Analysis of existing approaches with proposed approach (RCFD)

S. No.	Recognition approaches	Recognition accuracy rate (%)
1	Fractal Brownian motion and K-means classifier approaches [12]	88.80
2	Box-counting method (BCM) and double-threshold algorithm approaches [10]	94.61 and 93.23
3	Contourlets-based local fractal dimensions (CLFD) and Manhattan distance and the nearest neighbor classifier approaches [18]	97.80
4	Box-counting (BCM) and match score approaches [20]	92
5	Box-counting (BCM) and double-threshold algorithm approaches [21]	92
6	Differential box-counting method (DBC) and SVM classifier approaches [22]	97.9
7	Proposed method (RCFD) and DSL classifier approaches	98

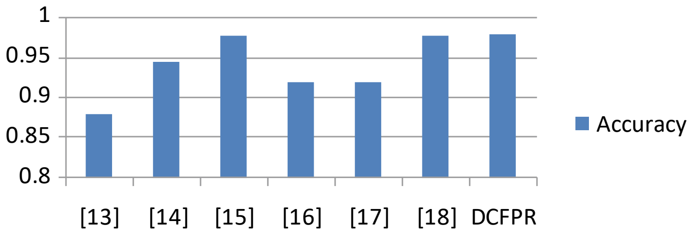


Fig. 13 Performance analysis of proposed recognition approaches with other existing approaches

DCFPR system is executed well in its proposed feature extraction approach RCFD, which is revealed by analyzing other existing approaches along with its recognition accuracy rate.

6 Discussion

The motive of this research is to bring effective PRS by emphasizing an innovative recognition approach that consists of RCFD feature extraction and DSL classifier approaches. Table 2 is exposed the admissible performance of the approaches used in this system and analyzed it with the performance of other existing recognition approaches in Table 3 to know the achievement attained in its improvement. Data, in the Fig. 13, conspicuously proved that this system attained improvement with the hold of 98% recognition accuracy rate is higher than the other approaches used in the existing biometric technologies. Thence, this proposed DSFPR system can assent as a good reliable way of recognition in biometric technology.

7 Conclusion

The proposed DCFPR system is designed with a novel feature extraction approach (RCFD) to extract the distinct unique features of preprocessed 2D-PROI contour images and that fed to the DSL classifier algorithm. DRCPR system is experimented on 400 2D-PROI images of POLYU database with 80% of training and 20% of testing images and achieved a better accuracy of 98% in the authentication process. Hence, it can be used as one of the good feasible ways of BAIS. However, this proposed system exposed more self-similarity values for several datasets that tend to lack recognition accuracy. Hence, future work will be extended with deep study in the feature extraction techniques to improve the recognition and identification performance of the system.

References

1. Gururaj J, Prashant B (2020) Biometric recognition of humans using scanned images of the retinal part of the eyes. Int J Future Gener Commu Network 13(1):1055–1071
2. Wayman JL (2001) Fundamentals of biometric authentication technologies. In: National Biometric Test Center, National Biometric Test Center, pp 1997–2000
3. Hussein IS, Sahibuddin SB, Sjarif NNA (2018) The fundamentals of unimodal palmprint authentication based on a biometric system: a review. Int J Adv Comput Sci Appl 9(11)
4. Ali MMH, Yannawar P, Gaikwad AT (2016) Study of edge detection methods based on palmprint lines. In: International conference on electrical, electronics, and optimization techniques

5. Humeau-Heurtier A (2019) Texture feature extraction methods: a survey. IEEE 7:2169–3536
6. Mokni R, Kherallah M (2016) Palmprint recognition through the fractal dimension estimation for texture analysis. Int J Biometrics 8(3–4):254–274
7. Raouia M, Hassen D, Monji K (2020) Deep-analysis of palmprint representation based on correlation concept for human biometrics identification. Int J Digital Crime Forensics, pp halshs-03147087
8. Shi X, Cheng HD, Hu L, Ju W, Tian J (2010) Detection and classification of masses in breast ultrasound images. Digital Signal Process 20(3):824–836
9. Ivanovici M, Richard N (2011) Fractal dimension of color fractal images. IEEE Trans Image Process 20(1)
10. Yui L, Zhang D, Wang K, Yang W (2005) Coarse iris classification using box-counting to estimate fractal dimensions. Pattern Recogn Soc (Elsevier) 38:1791–1798
11. Kisan S, Mishra S, Bhattacharjee G, Bansal R (2018) Analytical study on fractal dimension—a review. In: International conference on recent innovations in electrical, electronics & communication engineering (ICRIEECE), Bhubaneswar, India
12. Chang R-F, Chen C-J, Ho M-F (2004) Breast ultrasound image classification using fractal analysis. In: Proceedings of the Fourth IEEE symposium on bioinformatics and bioengineering (BIBE'04)
13. Ata MM, Elgamily KM, Mohemed MA (2020) Toward palmprint recognition methodology based machine learning techniques. Euro J Electr Eng Comput Sci 4(4)
14. Dandan Z, Xin P, Xiaoling L, Xiaojing G (2015) Palmprint recognition based on deep learning. In: ICWMMN proceedings
15. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521:436–444
16. Shen L, Margolies LR, Rothstein JH, Fluder E, McBride R, Sieh W (2019) Deep learning to improve breast cancer detection on screening mammography. Sci Reports 9:12495
17. Lozano-Diez A, Zazo R, Toledano DT, Gonzalez-Rodriguez J (2017) An analysis of the influence of deep neural network (DNN) topology in bottleneck feature based language recognition. PLOS ONE
18. Pan X, Ruan Q, Yanxia W (2008) Palmprint recognition using Contourlets-based local fractal dimensions. In: ICSP proceedings IEEE
19. Kumar T, Bhushan S, Jangra S (2021) An improved biometric fusion system of fingerprint and face using whale optimization. Int J Adv Comput Sci Appl (IJACSA) 12(1)
20. Farhan MH, George LE, Hussein AT (2014) Fingerprint identification using fractal geometry. Int J Adv Res Comput Sci Software Eng 4(1)
21. Patil PS (2015) IRIS classification based on fractal dimension box counting method. Int J Comput Appl 112(11):0975–8887
22. .Soni G, Vishwakarma S (2017) Face recognition using fractal dimensions (FD) based feature. Int J Eng Manage Res 7(5):140–146

Automatic Text Summarization of Konkani Texts Using Latent Semantic Analysis

Jovi D'Silva, Uzzal Sharma, and Chaitali More

Abstract Automatic text summarization involves extracting relevant details from the contents of input text documents for generating summaries. This area of Natural Language Processing is widely researched, especially with popular languages like English. There is a need to extend this work to less commonly spoken languages of the world. This paper presents a language-independent text summarization approach using Latent Semantic Analysis in Konkani language. Konkani is a low-resource language with limited language processing tools, stop-word list, etc. Latent Semantic Analysis (LSA) is an unsupervised algebraic method that finds latent semantic structures to be used for performing extractive text summarization. We examined well-known Latent Semantic Analysis-based sentence selection approaches on our dataset, constructed using books on Konkani folk tales written in Devanagari script. The results of the experiments indicated that LSA-based approaches can produce promising summaries, with the Cross method performing the best in most metrics.

Keywords Automatic text summarization · Latent semantic analysis · Konkani · Low-resource · Singular value decomposition · Extractive text summarization

1 Introduction

The number of documents on the Internet has grown exponentially in the last couple of years [1]. Reading many documents in order to determine if the contents are important to a reader or not is a daunting task. Text summarization systems could help

J. D'Silva (✉) · U. Sharma
Assam Don Bosco University, Guwahati, Assam 782402, India
e-mail: jovidsilva@gmail.com

U. Sharma
e-mail: uzzal.sharma@dbuniversity.ac.in

C. More
Fr. Agnel College of Arts and Commerce, Pilar, Goa 403203, India
e-mail: chaitali.more27@gmail.com; chaitali@fragnelcollege.edu.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_37

the readers by providing them with a summary of the contents [1]. The summary, thus generated, captures all the relevant details of the text and eliminates any redundant information [2–4]. This can help save a lot of a reader's time, as a reader can determine the importance of a document by just reading the summary.

The summaries generated by such automatic text summarization (ATS) systems can be classified into two types: extractive and abstractive [4, 5]. Extractive summaries are generated by selecting and extracting important sentences from an input document to form the summary. Abstractive summaries are generated in the way humans would summarize a document, i.e. understanding the matter conveyed in a document and providing a gist of the contents using words and phrases different from those found in the document. Creating abstractive summaries is a more complex process in comparison to creating extractive summaries. This is because, for producing abstractive summaries, the semantics need to be understood by a system along with finding the alternative words and phrases that convey the same meaning. However, for generating extractive summaries, a system has to find the important sentences and put them together. The relatively straightforward approach makes extractive summarization a popular choice [4, 5].

The methods used to develop ATS systems can be classified as language-dependent methods and language-independent methods [6]. Language-dependent methods use language-specific tools like language-specific dictionary for lemmatization, lexicons of words for stop-word removal and so forth [7]. However, a language-independent system does not depend on the usage of any such language-specific knowledge or pre-existing tools [7].

ATS is a widely researched area in Natural Language Processing (NLP). Most research pertains to extractive summarization, with English being the preferred choice of language because of its popularity. There is a need to extend this research to low-resource languages as well. We attempt to explore ATS in Konkani, a language that is spoken by a very small population of people along the west coast of India, called the Konkan Coast. As per the 2011 population census of India, only 2.2 million people speak Konkani [8]. The number of Konkani language speakers has been on a decline in recent years, and such research in a field like NLP will help boost its digital growth and its availability to the speakers and readers around the world. ATS systems can provide readers with summaries to help them find relevant documents without having to read them entirely. Konkani is a low-resource language which has limited language processing tools such as lemmatizers, stop-word lists and/or subject experts, which are readily available in popular languages like English. Therefore, we need to develop a language-independent approach to text summarization for low-resource languages to eliminate the dependency on any such tools.

In this paper, we present an automatic text summarization system for Konkani language texts based on Latent Semantic Analysis (LSA) using a language-independent approach [6, 9, 10]. It is an unsupervised algebraic method that finds unseen semantic structures of words or terms and sentences in a document [11–15]. LSA uses Singular Value Decomposition (SVD) to identify important concepts in a document and extracts relevant sentences to form a summary based on these concepts.

The major contributions of this study are as follows:

- It is the first attempt at studying the use of unsupervised Latent Semantic Analysis for extractive ATS of Konkani language folk tales.
- To examine the performance of popular sentence selection approaches in literature for the text summarization task using LSA and our Konkani dataset.
- To evaluate the summaries generated by the system against human-generated summaries using ROUGE toolkit.

The paper is organized as follows. The related work pertaining to LSA in text summarization is given in Sect. 2. The details of the Konkani dataset used for this experiment can be found in Sect. 3. The details of the methodology used in this experiment are given in Sect. 4. The four approaches used for sentence selection are described in Sect. 5. The experimental results and evaluations can be found in Sect. 6. Finally, the conclusion is given in Sect. 7.

2 Related Work

ATS is a widely researched area and has gained popularity in recent years. Some semantic-based methods used for text summarization are Latent Semantic Analysis, Explicit Semantic Analysis (ESA) and Semantic Role Labelling (SRL). LSA is commonly used semantic-based unsupervised approach used for extractive text summarization. It depicts semantics of text based on the observation of co-occurrence of words [9].

In 2001, Gong and Liu proposed the use of a standard IR method and LSA method for text summarization [11]. The IR method ranks sentences based on their relevance, and LSA determines the sentences that are semantically relevant. Both the methods choose highly ranked sentences that are different from one another. Steinberger and Jezek, in 2004, presented a generic text summarization method using LSA and evaluated the summaries using two LSA-based evaluation methods [12]. They found two drawbacks in the approach implemented by Gong and Liu and suggested a solution, i.e. the extraction of a single sentence score based on LSA along with variable dimensionality reduction. Murray et al. performed extractive summarization of multi-party meetings using LSA and used the ICSI Meetings Corpus [13]. They addressed the concerns of Steinberger and Jezek about the approach proposed by Gong and Liu and suggested that the best ‘n’ sentences be extracted rather than the best sentences in every topic. The value of ‘n’ could be established by the singular values. Their approach showed that dimensionality reduction was not bound to the length of a summary and multiple sentences from each topic could be selected. Ozsoy et al. implemented a generic text summarization approach based on LSA for generating extractive summaries of Turkish text documents [14]. They also proposed two novel LSA-based summarization methods. The Cross method proposed produced better results as compared to the other methods. Another work presented by Ozsoy et al. describes various LSA-based text summarization methods [15]. They implement the methods to perform extractive text summarization for Turkish and English texts and evaluated the performance of these methods.

In 2015, LSA-based approach was used for extractive text summarization of Kannada text documents [16]. Singular Value Decomposition (SVD) was used to determine sentence vector dimensions and aid in generating summaries. A hybrid technique for text summarization based on LSA and random indexing-based summarization method, which implicitly uses graph-based ranking approach, was presented in [17]. Sellers analysed the text summarization techniques, LSA and TextRank and evaluated the output. The evaluation showed that LSA produced better quality summaries as compared to TextRank [18].

Cagliero et al. used LSA for summarizing multilingual documents; the techniques, based on LSA and frequent itemset mining, are found to be the most effective for multilingual summarization [19]. They proposed a technique using the best features of both the approaches to overcome their individual shortcomings. Another approach for text summarization was presented in [20], using LSA in combination with diversity constraint using query-based ranking. They used Latent Semantic Indexing (LSI) for sentence selection and ranking. They computed the diversity between the selected sentences, presuming that maximum diversity in sentences is an indicator of good quality summary. Bilel et al. presented a new Automatic Arabic Text Summarization (AATS) system which works by guessing important keywords and constructing a document summary [21]. The work presented in [22] demonstrates extractive summarization of text documents with LSA topic modelling and 'Term-Frequency Inverse Document Frequency' (TFIDF) keywords extractor for every sentence in a document. They also used 'Bidirectional Encoder Representations from Transformers' (BERT) model for sentence encoding for the retrieval of positional embeddings of topic word vectors.

To summarize, LSA is an effective language-independent text summarization approach that has been effectively used to summarize texts in English, Turkish, Arabic and Kannada [6, 9, 10]. The work done so far has used documents from different genres such as news, Wikipedia, meetings and scientific articles. We extend this approach to the genre of literature using folk tales in Konkani and examine the use of LSA in a low-resource language for automatic text summarization.

3 Dataset

The dataset used for this experiment was constructed by the authors of this paper using 5 unique Konkani books on folk tales written by various Konkani authors. Folk tales have a prime significance in local cultures as these stories have a moral or good teaching that is passed to future generations through the medium of stories. The dataset is composed of 71 stories written in Devanagari script. It belongs to the genre of literature, and there was no such dataset available at the time this work was in progress.

The books were collected, and the contents were converted to a digital form for further processing. The dataset has a total of 9849 sentences with each story having an average of 138 sentences and 1155 words. The longest story in the dataset has 2852 words, and the shortest story has 520 words. For each of the 71 stories, two independent language experts generated 300-word ‘gold-standard’ abstracts which were used for evaluating the extractive summaries generated by the system. The threshold of 300 words was established on consulting the language experts and also considering the length of the shortest story, i.e. 520 words. The detailed process of dataset construction is available in [23].

4 LSA for Automatic Text Summarization

Latent Semantic Analysis (LSA) is an algebraic technique that tries to identify latent or hidden semantic structure from an input document which can extract important sentences to form a summary. LSA is completely unsupervised and does not rely on any training data, thus making LSA an unsupervised semantic-based extractive text summarization approach. The input documents are comprised of sentences, and these sentences are made up of words or terms. The LSA approach uses SVD to determine the relationship between sentences and the terms in an input document. It does this by modelling the input document as a Term-Document Matrix, following which concepts are derived from this matrix using SVD. Using these derived concepts, important sentences are selected from the input document to form a summary [9–15]. The summarization process can be expressed in the following four detailed steps and is illustrated in Fig. 1.

4.1 Pre-processing

Every input document goes through a phase called ‘tokenization’, where the text document is broken down into smaller units or sentences. The sentences are cleaned of any punctuation, although punctuations are retained in the final output summary.

4.2 Term-Document Matrix Formation

The input document is represented as a Term-Document Matrix, where the columns are the sentences of the input document and rows are terms (words) present in the document. The cell values used for the matrix depend on the approach used to represent these values.

We examined two approaches to populate the cell values; they are ‘Term-Frequency Representation’ and ‘Binary Representation’. In the Term-Frequency

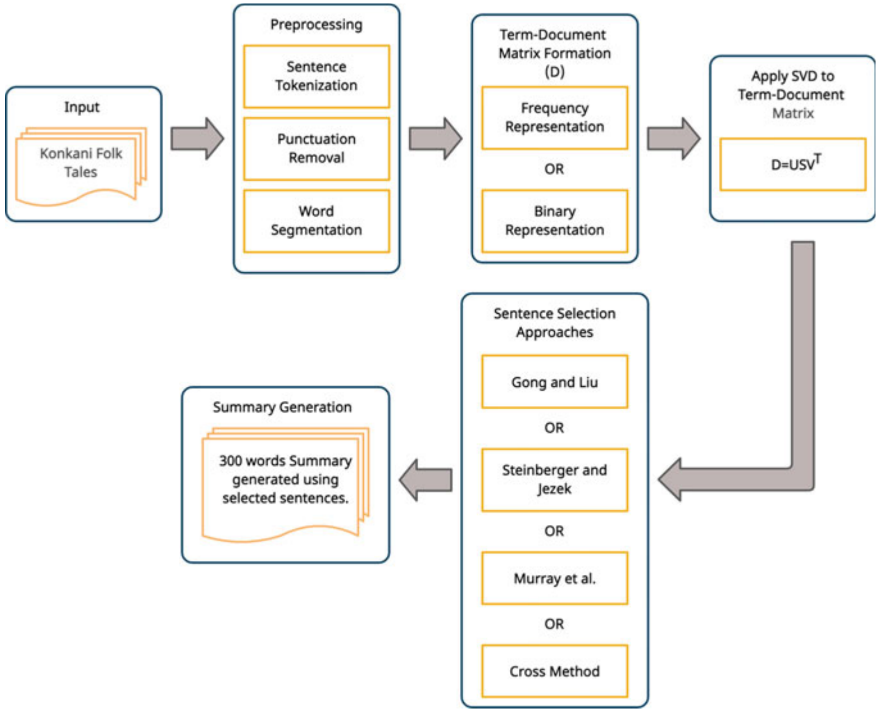


Fig. 1 Summarization process using LSA

Representation, the cells contain the frequency of occurrence of the term in the sentence. Binary Representation marks a term that occurs in a sentence as '1' and a term that does not occur as a '0'. Both approaches produce sparse matrices of order $m \times n$, where 'm' is the number of terms and 'n' is the number of sentences in the matrix [11, 12].

4.3 Apply SVD to Term-Document Matrix

In this step, SVD is applied depending on the Term-Document Matrix formation approach selected in the previous step. SVD is an unsupervised algebraic approach that models the relationship between the terms present in the document and the sentences. Given a matrix D as input, it produces three new matrices as output [11–13]. This is expressed mathematically in Eq. (1).

$$D = USV^T \tag{1}$$

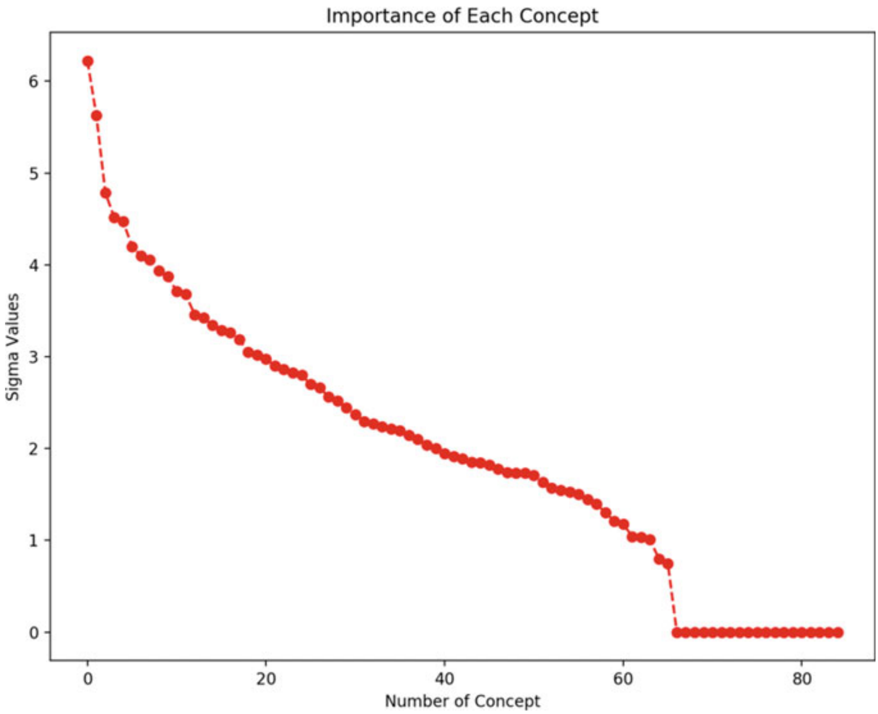


Fig. 2 Importance of each concept

where D is an input Term-Document Matrix ($m \times n$), Matrix U is *terms* $\times$ *extracted concepts* ($m \times n$) and is a column-orthonormal matrix, Matrix S represents the scaling values as a diagonal descending matrix ($n \times n$), and Matrix V is *sentences* $\times$ *extracted concepts* ($n \times n$) and is an orthonormal matrix. SVD takes the number of concepts to be extracted as an input hyperparameter [24]. We set this value to the total number of sentences in a document. Matrix S represents sigma values which highlight the importance of each of the concepts. The first concept is the most important, and as we increase the number of concepts, their importance will be lesser. Some selection approaches, such as Steinberger and Jezek [12], reduce the number of dimensions or concepts before sentence selection. The plot of sigma values versus the number of concepts for a document with 85 sentences is shown in Fig. 2.

4.4 Sentence Selection Approaches for Summary Formation

Once SVD has been calculated on the Term-Document Matrix, we use the results to select important sentences for generating the summary. We use four approaches for sentence selection. The details of these approaches are presented in Sect. 5.

5 Sentence Selection Approaches

Sentence selection is performed using the results of Singular Value Decomposition. In this section, we give a detailed description of the four sentence selection methods used in our experiment.

5.1 Gong and Liu [11]

We implement the sentence selection algorithm used by Gong and Liu [11]. The input document is represented in matrix form, and the SVD values are calculated. Thereafter, the matrix with the extracted *concepts x sentences* is used for the selection of relevant sentences. In matrix V^T, the rows represent concepts ordered from highest relevance to the lowest relevance and the columns represent sentences. The matrix values represent the relationship between the concepts and the sentences. A higher value of an element in the matrix indicates that a sentence has a better relation to the concept. According to Gong and Liu's algorithm, one sentence is selected from the most crucial concept and the next sentence is selected from the second most crucial concept. The process is repeated in this manner until a desired number of sentences are gathered.

This method has some drawbacks mentioned as follows: (i) If the desired number of sentences to be collected is too high, then sentences with not so relevant concepts get selected and (ii) only one sentence is selected from each concept; some very critical concepts may have multiple relevant sentences which could be skipped from selection.

5.2 Steinberger and Jezek [12]

In Steinberger and Jezek approach, the length of each sentence vector in matrix V is computed [12]. The values of the matrix S that are greater than or equal to a threshold value are considered for further computations. This step also reduces the dimensions of the matrix V. The threshold is computed by dividing the highest sigma value by 2. The square of each sigma value is multiplied with each of the corresponding squared values from the 'concept' column; this is done to favour important concepts. Thereafter, the length of each sentence is computed by summing the values across the concepts for that sentence and computing the square root of that sum. Sentence lengths are used as a saliency score, and the sentences which have the highest scores are then selected to generate the summary. This approach overcomes the drawbacks of the Gong and Liu approach; the sentences associated with all the important concepts are selected, and it also permits the selection of multiple sentences from an important concept.

5.3 Murray et al. [13]

Murray et al. improved on Gong and Liu's approach for sentence selection [13]. The algorithm, thus used, could select more than one sentence from the relevant concepts of a document, positioned in the first row of the V^T matrix. Multiple sentences related to a very important concept can be selected even if their value in the matrix is not the highest. The number of sentences to be selected is determined by the S matrix. This value is computed by obtaining the percentage of a related singular value corresponding to a concept with the sum of all the singular values for every concept.

5.4 Cross Method, Ozsoy et al. [15]

Ozsoy et al. extended the sentence selection approach used by Steinberger and Jezek [12] and devised the Cross method [14]. This method introduces a pre-processing step before the sentence selection step to eliminate the overall impact of the sentences related to a concept, but not the core sentences of the concept. The average sentence score is computed for each concept and represented by the rows of the V^T matrix. Thereafter, the values of the matrix elements less than or equal to this score are set to zero. This step ensures that the sentences which are more related to a concept are retained and the ones lesser related are eliminated. Thereafter, the total length of every sentence vector, given by the columns of matrix V^T, is computed by adding the concept values for that sentence. The number of concepts to be used is provided by the user. In the absence of this input, all the extracted concepts are applied. Thereafter, the longest sentence vectors are selected to form the output summary.

6 Experimental Evaluation

We have implemented language-independent automatic text summarization systems for generating extractive summaries of Konkani texts using LSA. The implemented systems produced extractive summaries for each of the 71 folk tales of the Konkani dataset, and these summaries were compared against the 'gold-standard' human-generated summaries. The output summaries were considered being good if the word composition of the system-generated summaries closely matched that of the human-generated summaries. ROUGE toolkit was used to evaluate the system-generated summaries using ROUGE-1, ROUGE-2 and ROUGE-L measures [25]. ROUGE uses N-gram statistics and works well with human evaluations.

ROUGE accounts for overlap of unigrams, bigrams and 'Longest Common Subsequence' (LCS) in the human-generated summaries and the system-generated summaries. ROUGE-1 gives an account of unigrams, ROUGE-2 gives an account of

Table 1 ROUGE-1 values

System name	Precision	Recall	*F*-Score
LSA-Frequency-Gong	0.31952	0.31987	0.31967
LSA-Frequency-Steinberger	0.31545	0.31555	0.31548
LSA-Frequency-Murray	0.31705	0.31696	0.31698
LSA-Frequency-Cross	0.32574	0.32489	**0.32529**
LSA-Binary-Gong	0.31927	0.31956	0.31940
LSA-Binary-Steinberger	0.31725	0.31732	0.31726
LSA-Binary-Murray	0.32039	0.32050	0.32042
LSA-Binary-Cross	0.32828	0.32748	**0.32786**

Table 2 ROUGE-2 values

System name	Precision	Recall	*F*-Score
LSA-Frequency-Gong	0.08039	0.08048	0.08042
LSA-Frequency-Steinberger	0.08140	0.08141	**0.08140**
LSA-Frequency-Murray	0.08050	0.08048	0.08049
LSA-Frequency-Cross	0.07530	0.07516	0.07523
LSA-Binary-Gong	0.07970	0.07978	0.07974
LSA-Binary-Steinberger	0.07991	0.07990	0.07990
LSA-Binary-Murray	0.08117	0.08121	**0.08118**
LSA-Binary-Cross	0.07542	0.07529	0.07535

bigrams, and ROUGE-L accounts for LCS scores. These measures help in analysing the fluency of the system-generated summary with increasing text granularity. The values obtained for ROUGE-1, ROUGE-2 and ROUGE-L are presented in Tables 1, 2 and 3, respectively. It can be noted that the values of ROUGE-1 are greater than that of ROUGE-2 by observing Tables 1 and 2; this is because an identical term can be present in both the metrics but it may not be true conversely. We also notice the values of ROUGE-L are more than the values of ROUGE-2, since ROUGE-L keeps track of the words appearing in order in a sentence but not necessarily appearing

Table 3 ROUGE-LCS values

System name	Precision	Recall	*F*-Score
LSA-Frequency-Gong	0.31218	0.31252	0.31233
LSA-Frequency-Steinberger	0.30760	0.30769	0.30762
LSA-Frequency-Murray	0.30962	0.30954	0.30956
LSA-Frequency-Cross	0.32072	0.31988	**0.32028**
LSA-Binary-Gong	0.31130	0.31159	0.31143
LSA-Binary-Steinberger	0.30903	0.30911	0.30905
LSA-Binary-Murray	0.31286	0.31298	0.31290
LSA-Binary-Cross	0.32263	0.32185	**0.32222**

consecutively. Thus, ROUGE-L includes the longest common n-grams appearing in sequence. The systems with the highest scores are highlighted in bold text in the tables.

'Precision' and 'Recall' provide the measures of the overlap between the system-generated summaries and the human-generated summaries. 'Precision' scores are indicative of the extent to which the contents of the system-generated summary are relevant. 'Recall' scores are indicative of the extent to which the contents of the human-generated 'gold-standard' summaries are represented in the system-generated summaries. '*F*-Score' combines the Precision and Recall scores and gives a single score. The scores obtained are in the range of 0 to 1. A score of '0' indicated that there is no overlap between the system-generated summary and human-generated reference summary, whereas a score of '1' suggests that there is a very strong overlap of content between the two summaries.

From the results depicted in the tables, we observe that when frequency mode was used to construct the Term-Document Matrix, the Cross method outperformed the other methods in ROUGE-1 and ROUGE-LCS metrics and this is consistent with what has been reported by Ozsoy et al. [15]; it must be noted that this is to be expected as ROUGE-1 and ROUGE-LCS metrics are highly correlated. However, for the ROUGE-2 metric, Steinberger's approach scored the highest, indicating that the sentences chosen had higher number of bigram overlap. When binary mode was used for Term-Document Matrix construction, we again observe that for ROUGE-1 and ROUGE-LCS, Cross method outperformed the other methods but lost out to Murray's approach in the ROUGE-2 metric. When considering all the metrics and Term-Document Matrix construction modes, Cross method achieved the highest score in a number of metrics. LSA-based techniques are completely unsupervised and do not make use of any language-dependent resources, but can produce promising summaries.

7 Conclusion

In this paper, we implemented a language-independent automatic text summarization system for generating extractive summaries of Konkani texts using LSA. LSA, being unsupervised, does not require any training data. LSA makes use of SVD to derive latent concepts of importance from a document the results of which are used for sentence selection to perform extractive text summarization. We examined well-known sentence selection approaches in literature, namely Gong and Liu, Steinberger and Jezek, Murray et al. and Ozsoy et al., on our dataset comprising of folk tales in Konkani. From the evaluations conducted using the metrics in the ROUGE toolkit, the results show that the Cross method, proposed by Ozsoy et al., performed the best in ROUGE-1 and ROUGE-LCS metrics except for ROUGE-2 metric. Unsupervised language-independent models, such as LSA, can be extended to other low-resource languages such as Konkani, with limited language processing tools and stop-word list, and yet produce promising summaries.

References

1. Hobson S, Dorr B, Monz C, Schwartz R (2007) Task based evaluation of text summarization using relevance prediction. Inf Process Manage 43(6):1482–1499. https://doi.org/10.1016/j.ipm.2007.01.002
2. Ling X, Jiang J, He X, Mei Q, Zhai C, Schatz B (2007) Generating gene summaries from biomedical literature: a study of semistructured summarization. Inf Process Manage 43(6):1777–1791. https://doi.org/10.1016/j.ipm.2007.01.018
3. Lloret E, Palomar M (2012) Text summarisation in progress: a literature review. Artif Intell Rev 37:1–41. https://doi.org/10.1007/s10462-011-9216-z
4. Andhale N, Bewoor LA (2016) An overview of text summarization techniques. In: International conference on computing communication control and automation (ICCUBEA).https://doi.org/10.1109/iccubea.2016.7860024
5. Moratanch N, Chitrakala S (2017) A survey on extractive text summarization. In: IEEE international conference on computer, communication and signal processing (ICCCSP), pp 1–6 (10–11 Jan 2017). https://doi.org/10.1109/icccsp.2017.7944061
6. D'Silva J, Sharma U (2019) Automatic text summarization of Indian languages: a multilingual problem. J Theor Appl Inf Technol 97(1). ISSN: 1992-8645
7. Saleh AA, Weigang L (2017) Language independent text summarization of western European languages using shape coding of text elements. In: 2017 13th International conference on natural computation, fuzzy systems and knowledge discovery (ICNC-FSKD), pp 2221–2228. https://doi.org/10.1109/FSKD.2017.8393116
8. Statement—2 (2011) Distribution of population by schedule and other languages India, States and Union Territories—2011. Office of the Registrar General & Census Commissioner, India, Ministry of Home Affairs, Government of India. Available: https://censusindia.gov.in/2011Census/Language-2011/Statement-2.pdf. Accessed 15 Nov 2020
9. El-Kassas WS, Salama CR, Rafea A, Mohamed HK (2021) Automatic text summarization: a comprehensive survey. Expert Syst Appl 165:113679
10. Sarkar K (2014) Multilingual summarization approaches. Computational linguistics: concepts, methodologies, tools, and applications, pp 158 (January 2014). https://doi.org/10.4018/978-1-4666-6042-7.ch009
11. Gong Y, Liu X (2001) Generic text summarization using relevance measure and latent semantic analysis. In: Proceedings of the 24th annual international ACM SIGIR conference on research and development in information retrieval, pp 19–25
12. Steinberger J, Jezek K (2004) Using latent semantic analysis in text summarization and summary evaluation. In: Proceedings of ISIM. Apr 4: 93–100
13. Murray G, Renals S, Carletta J (2005) Extractive summarization of meeting recordings. In: INTERSPEECH
14. Ozsoy M, Cicekli I, Alpaslan F (2010) Text summarization of Turkish texts using latent semantic analysis. In: Proceedings of the 23rd International conference on computational linguistics (Coling 2010), pp 869–876 (Aug 2010)
15. Ozsoy MG, Alpaslan FN, Cicekli I (2011) Text summarization using latent semantic analysis. J Inf Sci 37(4):405–417
16. Geetha JK, Deepamala N (2015) Kannada text summarization using latent semantic analysis. In: 2015 International conference on advances in computing, communications and informatics (ICACCI), pp 1508–1512. https://doi.org/10.1109/ICACCI.2015.7275826
17. Chatterjee N, Yadav N (2019) Hybrid latent semantic analysis and random indexing model for text summarization. In: Fong S, Akashe S, Mahalle P (eds) Information and communication technology for competitive strategies. Lecture notes in networks and systems, vol 40. Springer, Singapore. https://doi.org/10.1007/978-981-13-0586-3_15
18. Sellers J (2019) Evaluation of LSA and TextRank methods for automatic text summarization. Dissertation, University of Washington, Seattle, WA

19. Cagliero L, Garza P, Baralis E (2019) ELSA: a multilingual document summarization algorithm based on frequent itemsets and latent semantic analysis. ACM Trans Inf Syst (TOIS) 37(2):1–33. https://doi.org/10.1145/3298987
20. Mandal S, Singh GK (2020) LSA based text summarization. Int J Recent Technol Eng (IRTE) 9(2):150–156. ISSN: 2277-3878
21. Elayeb B, Chouigui A, Bounhas M et al (2020) Automatic Arabic text summarization using analogical proportions. Cogn Comput 12:1043–1069. https://doi.org/10.1007/s12559-020-09748-y
22. Gupta H, Patel M (2021) Method of text summarization using LSA and sentence based topic modelling with Bert. In: 2021 International conference on artificial intelligence and smart systems (ICAIS), pp 511–517. https://doi.org/10.1109/ICAIS50930.2021.9395976
23. D'Silva J, Sharma U (2019) Development of a Konkani language dataset for automatic text summarization and its challenges. International Journal of Engineering Research and Technology12(10):1813–1817. ISSN 0974-3154. https://doi.org/10.5281/zenodo.5531954
24. Pedregosa F et al (2011) Scikit-learn: machine learning in python. J Mach Learn Res 12:2825–2830
25. Lin CY (2004) ROUGE: a package for automatic evaluation of summaries. In: Proceedings of the workshop on text summarization branches out (WAS 2004), Barcelona, Spain, pp 74–81

Supervised Automatic Text Summarization of Konkani Texts Using Linear Regression-Based Feature Weighing and Language-Independent Features

Jovi D'Silva and Uzzal Sharma

Abstract Automatic summarization of text documents is a widely researched domain in natural language processing. A lot of research is carried out on the most commonly spoken languages in the world. Automatic text summarization needs to be explored to include some of the less popular languages in the world to help sustain such languages and promote their use. A language-independent summarization system that can be effortlessly extended to other such languages, which could have a limited number of resources to carry out such research is required. In this paper, we examine the efficiency of supervised linear regression models for the performing single document extractive automatic text summarization on Konkani language folk-tales dataset. We use 13 language-independent features and linear regression models to learn feature weights. These weights are then used to calculate a sentence's score; top ranking sentences are then chosen for summary generation. We employ a k-fold evaluation strategy to evaluate the system-generated summary against a human-generated summary using ROUGE evaluation toolkit. Additionally, we also evaluate the use of L1 and L2 regularization on the summarization task. The work represents early attempts in automatic text summarization pertaining to Konkani language, and the dataset employed in these experiments is unique and devised particularly to facilitate research in this domain. The language-independent features used can be readily extended to other low-resource languages. The systems implemented in this work performed better as compared to an unsupervised system based on k-means approach and also beat the baseline systems.

Keywords Supervised machine learning · Text summarization · Konkani · Regression · Extractive · Natural language processing · Low resource · Language-independent features

J. D'Silva (✉) · U. Sharma
Assam Don Bosco University, Guwahati, Assam 782402, India
e-mail: jovidsilva@gmail.com

U. Sharma
e-mail: uzzal.sharma@dbuniversity.ac.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_38

1 Introduction

Automatic summary generation has gained tremendous significance over a period of years, mainly because of the enormous amount of information available on the Web today [1]. It is crucial to provide users with a gist of the document so that one can determine if they need to invest their time reading an entire article or not. Text summarization condenses the contents of a long text document into a concise outline that preserves the meaning of the original document [2, 3]. The field of text summarization has gained immense popularity in Natural Language Processing (NLP). Hence, when the summarization of texts is done automatically by a computer system, it is termed as 'Automatic Text Summarization' (ATS). The summaries produced by ATS systems can be categorized as abstractive summaries or extractive summaries. Abstractive summaries are those, wherein the summary is generated by capturing the meaning of the information conveyed in the text document and then generating a summary that may be reworded by the summarizer. Extractive summaries are the ones that are generated by picking up relevant sentences or phrases in their original form the source document to generate the summary [4, 5].

There is a growing need to explore the field of ATS to include the less-popular languages in ATS. Languages that are spoken by most of the global population, like English, have been the primary candidates for such research. However, the Internet houses articles, books, and other work in many other languages too. Therefore, there is a need for automatic text summarization tools that can effortlessly summarize documents written in other, less popular languages, as well. The methods that are used to implement such text summarization tools can be classified as language dependent or language-independent methods [6]. Language-dependent methods rely on tools related to the language, like lexicons of words for stop-word elimination, language-specific dictionary for lemmatization, and so on [7]. A language-independent system does not depend on any such language-specific tools or knowledge [7].

In this paper, we propose a language-independent method using supervised machine learning approach for generating extractive text summaries by using linear regression and language-independent features. The method was used to experiment on a dataset of Konkani language documents, specifically designed for ATS research [8]. An attempt was made to extend the domain of ATS to a low-resource language, Konkani, which is a language spoken by a minority population of India. A language with a limited data, tools, language experts, and/or speakers is considered as a low-resource language. This was the primary motivation when using language-independent features for the experiment. Konkani fits this criterion since it has not been a preferred language for research in NLP.

Konkani language is spoken along the Konkan coast, that lies on the west coast of India and is spoken by only 2.2 million people in India as per the census of India data of 2011 [9], which amounts to just about 0.19% of the total population of the country. These numbers have rapidly declined over the previous decade and continue to decline. Hence, it is very crucial, given the current status of this language, to help its digital growth by providing readers with digital versions of Konkani texts and encouraging more people to read and contribute their literary knowledge of the language. Konkani ATS tool could help in boosting the digital growth of the language by summarizing text documents to its readers. Konkani is one of the oldest languages of the country, with the earliest scriptures dating back to 1187 A.D. [10]. This paper illustrates some of the initial attempts at including a low-resource language Konkani into the field of ATS.

The proposed technique uses linear regression algorithms, ordinary least squares, ridge regression, and Least Absolute Shrinkage and Selection Operator (LASSO) regression, for determining a sentence score and producing a summary of top ranking sentences. We use 13 language-independent features to represent the sentences of a text document. Traditionally, linear regression algorithms function by evaluating the values of multiple independent variables and then predicting the value of a dependent variable. The linear regression algorithms learn a set of coefficients to make a prediction as a weighted sum. In this experiment, we use these coefficients as feature weights. These feature weights are then used for calculation of ranking scores of sentences. The sentences are then selected to form a summary based on these ranks.

The major contributions of this study are as follows:

- It is the first attempt at studying the use of supervised linear regression algorithms for sentence scoring-based extractive ATS of Konkani language folktales.
- To examine the use of language-independent features for a low-resource language, Konkani.
- To examine the effects of L1 and L2 regularization on the text summarization task using regression.
- To compare and evaluate, ordinary least squares, ridge regression, and Least Absolute Shrinkage, and Selection Operator (LASSO) regression for text summarization.
- To evaluate the summaries generated by the system against the lead baseline, modified Luhn's heuristic baseline, and human-generated summaries using ROUGE toolkit.

The paper is organized as follows; the related work is given in Sect. 2. Section 3 highlights the dataset details. Section 4 gives the idea of the overall approach followed for summarization. Section 5 gives the details of the algorithms. Section 6 describes the choice of regressands. Section 7 outlines the features used for feature extraction. Section 8 recounts the methodology. Section 9 presents the results and discussion followed by the conclusion in Sect. 10.

2 Related Work

In the year 2009, Ulrich et al. proposed a Machine Learning (ML) approach based on regression for the summarization of emails. They showed that classifiers, based on regression, yielded better output compared to binary classifier [11]. Esther Hannah et al. presented an automatic text summarizer designed to produce summaries like humans. They generated a model that approximated a relation by utilizing multivariate statistical technique [12].

Xie and Liu illustrated an innovative approach to provide extractive summaries of lengthy meetings with supervised learning technique using regression [13]. Gupta and Lehal proposed an approach for the summarization of Punjabi language news articles by applying mathematical regression. It aided in the estimation of "text feature weights." They considered statistical and linguistic features while determining the relevance of a sentence [14].

Aristoteles et al. proposed using binary regression techniques to perform text feature weighting on Indonesian text [15]. Kumar et al. presented a technique for extractive summary generation using regression. They used ten distinct features, and weights were assigned to features using "weight learning method" [16]. Dlikman and Last explored and evaluated the performances of several ML algorithms when applied to perform extractive text summarization. They introduced an innovative technique based on a "similarity score" between a prospective sentence and the "benchmark summaries" [17, 18].

Oliveira et al. illustrated a novel regression-based methodology using "Integer linear programming (ILP)" for the summarization of single document. This proposal is based on the premise that "no single summarization technique can achieve high performance for all input articles." This new regression-based technique uses ILP for the generation of multiple probable summaries for every document. Thereafter, a regression method was applied to choose the summary that was the most "informative" of the lot [19].

Zopf et al. underlined that sentence regression technique determines the significance of every sentence with the help of "learned utility scores." Thereafter, the scores were used to pick sentences from the original document under a greedy selection strategy. They proposed that learning the prediction of "ROUGE precision scores" led to getting enhanced output [20]. Malallah and Ali built a mathematical model for weight estimation between independent and dependent variables by utilizing linear regression technique for multi-document summarization task. They used TAC 2011 dataset comprising English language documents and the results were evaluated using ROUGE [21].

Rezaei et al. applied supervised machine learning algorithms, like regression and classification, to extractive summarization of Persian news documents. They depicted the sentences of a document as feature vectors and assigned ranks to the sentences by considering the overlap with the "golden summaries" written by humans. The algorithms then learned the prediction of these ranks that were to be assigned to sentences of any document provided as input [22]. Cagliero and Quatra used regression for extracting highlights from scientific papers. The regression-based model trained on various features that described similarity between highlights and candidate sentences. The model predicted the similarity between the highlights and the sentences in the paper to be annotated [23].

El-Kassas et al. highlighted and compared various ATS methods. They noted that simple regression-based machine learning models achieve better results compared to other models used for summarization [24, 25].

To summarize, extractive summarization has been effectively used for text summarization. Linear regression-based methods have also been used for text summarization based on feature weighing [12]. Most research has been done on popular languages such as English. We use folktales written in Konkani for our experiment, and Konkani being a low-resource language, tools and datasets were not easily available. A ML system based on unsupervised k-means clustering algorithm using the same Konkani dataset has been presented in [26]. Another approach using supervised ML technique for extractive ATS using linear and nonlinear classification algorithms is presented in [27]. There is a need to expand the research to other such low-resource languages. Thus, we propose the use of language-independent features, and we employ linear regression algorithms for feature weighing-based sentence scoring. Further, we examine the impact of L1 and L2 regularization on the text summarization task. The summaries thus generated are evaluated against human-generated summaries.

3 Dataset

The dataset employed for this research using regression methods was specifically compiled to carry out research pertaining to ATS in Konkani literature [8]. The dataset comprises Konkani language folktales compiled from five unique books written by various authors. Folktales are short stories that have cultural significance with a lesson or a moral and were traditionally passed on verbally from a generation to another. “Konkani” is the primary language spoken along the Konkan Coast of India and has not been a popular language of choice for research in ATS. The motivation for producing a dataset from scratch was that there was no existing dataset in Konkani language literature at the time this study was undertaken. Hence, the dataset is unique, and the stories in this dataset are compiled from five rare books that were also difficult to procure. It includes 71 stories that are written in Devanagari script [8]. The dataset comprises 9849 sentences, and every story has an average of 138 sentences and 1155 words. The longest story in the dataset comprises 2852 words, and the shortest story comprises 520 words.

Two Konkani language experts manually generated “gold-standard” summaries for each of the 71 stories in the dataset, which were used for evaluation of the quality of the system-generated summaries. Apart from this, a new benchmark was presented comprising extractive summaries of 300 words composed using the sentences manually marked as relevant to include in an extractive summary by the same language experts. Every sentence in the documents was annotated by the language experts. This labeled data were used for training the ML algorithms. The summaries produced by the system and the human-annotated benchmark are extractive; these were then evaluated against the pair of “gold-standard” summaries to verify how closely they matched. The word limit of 300 words for summary length was chosen in consultation with the language experts, as it would fill up a page. Also, the length of the shortest story is 520 words; keeping this mind, 300 words limit seems apt for the summary length. The details of the dataset construction can be found in [8].

4 Approach

We use a supervised ML approach, where we use linear regression algorithms to learn feature weights given the feature vectors and regressands in the training instances. The algorithms used, especially with regularized algorithms, are affected by the scale of the features. Thus, before the training, all the features are scaled between 0 and 1. The regressands used are "1.0" and "0.0," which indicate if a sentence is to be included in the system-generated summary by marking each sentence with "1.0" and "0.0" if it is not to be included in the summary.

We make use of three linear regression algorithms: ordinary least squares, ridge regression, and LASSO regression. Linear regression algorithms use feature weights, also known as coefficients, to make a prediction as a weighted sum. Regularization is a means of placing a penalty on complex models. In terms of linear regression, it shrinks the coefficients or feature weights. Ordinary least squares algorithm does not perform any form regularization, but ridge regression performs L2 regularization which only shrinks feature weights and prevents over-fitting. LASSO performs L1 regularization which not only shrinks the feature weights but also can perform feature selection by setting some feature weights to exact zero, effectively negating their presence. The feature weights learnt by each of the three algorithms are then used to calculate a sentence score; the top scoring sentences are then selected to form the final summary.

Linear regression algorithms based on ML have shown to achieve better results compared to other models for summarization tasks [24]. Also, linear regression algorithms are easier to train, show better performance when generalizing new and unseen data and are not prone to over-fitting.

5 Regression Models for Text Summarization

Scikit Learn, a ML library, was used for implementing the algorithms [28]. Traditionally, in linear regression technique, the values of multiple independent variables are used for the estimation of the value of a dependent variable. These regression algorithms learn a set of coefficients to make predictions as a weighted sum. In this experiment, we use these coefficients as feature weights.

Our dataset contains 13 language-independent features. The model learns the weights of these 13 features. The weights indicate the relevance of each feature. The weighted sum of features is used as a sentence score, with the top sentences being selected to form the summary.

The three linear regression algorithms used for the experiments are ordinary least squares, ridge regression, and least absolute shrinkage and selection operator (LASSO) regression.

5.1 Ordinary Least Squares

This method constructs a model for the prediction of unknown variables that minimizes the sum of the squared errors between the predicted value and the observed values. Ordinary least squares algorithm does not perform any form of regularization. Mathematically, it is illustrated by Eq. (1).

$$L = \min_{w} \| X_w - y \|_2^2 \tag{1}$$

where X is training data, y is regressands, $w = (w_1, \ldots, w_p)$ are the coefficients, and L is the objective function to minimize.

5.2 Ridge Regression

Ridge regression solves some of ordinary least squares' issues by placing a penalty on the coefficients' size [28]. The equation is given by Eq. (2).

$$L = \min_{w} \| X_w - y \|_2^2 + \alpha \| w \|_2^2 \tag{2}$$

where $\alpha \geq 0$ is the complexity parameter that regulates the degree of "shrinkage," the larger the "α" value, greater is the shrinkage. Ridge regression uses L2 regularization which is given by $\alpha \| w \|_2^2$. In Scitkit Learn, we set the "α" hyperparameter to 0.08 [28].

5.3 LASSO Regression

LASSO regression, unlike ridge regression, does not take the square of the coefficients. It uses L1 regularization which can result in zero coefficients. So, LASSO not only puts a penalty on the coefficients but also aids in selecting important features. Mathematically, it can be represented by Eq. (3).

$$L = \min_{w} \|X_w - y\|_2^2 + \alpha\|w\|_1 \quad (3)$$

where "α" is a constant, and $\alpha\|w\|_1$ represents L1 regularization. In Scikit Learn, this is set as a hyperparameter with the value 0.001 [28].

For text summarization, the linear regression model helps in selecting important sentences by assigning each sentence a score. These scores are calculated by a weighted sum using each sentence's feature vector and the feature weights which are learnt by the linear regression algorithm. They can perform feature weighing or finding the importance of features, where the importance of a particular feature increases with its weight.

6 Choice of Regressands

These are the target values to be predicted by the system. They are continuous values in case of regression models although in our case we do not use the regressand. It is, however, crucial for building regression models, and it is the choice of the regressands which has to be predicted by the models. Most of the recent work attempted to predict ROUGE recall scores of individual sentences and some others suggested the use of "Cosine scores" that range between 0.0 and 1.0 [13, 17, 20, 21]. The regressand value is set to "1" or "0" depending on whether the sentence is present in the manual summary or not [12]. In our case, we use the values 1.0 or 0.0, which are assigned by subject experts based on if the sentences are significant or not with respect to inclusion in the system summary [12, 16].

7 Feature Extraction

We used of 13 features to represent the sentences. These features were also used in the supervised ML learning approach presented in [27]. The features chosen are language independent and are structure based or vector based. The features are language independent since they do not use any language-dependent resources nor do they use any language-specific knowledge but use statistics [29]. These statistics are calculated by considering characters, words, sentences in a document and can be calculated for a document in any language, i.e., counting the number of sentences or words or characters in a document for any language.

The following are some notations used in the equations stated below; "T" represents a source document. "P" is a sentence from a document. The position of sentence "P" in document "T" is denoted by "n." The number of times a term "m" appears in a document is given by "$tf(m)$." The number of sentences in a document "T" is depicted by "s," and "w" represents the total number of words in a sentence, and "c" is the number of characters in a sentence.

- **LOC_S**: Given by Eq. (4), it measures the position of a sentence to the start of a document [30].

$$\text{LOC\_}S = \frac{1}{n} \tag{4}$$

- **LOC_E**: The position of a sentence with respect to the end of a document, illustrated as "n" [31], given by Eq. (5)

$$\text{LOC\_}E = n \tag{5}$$

- **LOC_B**: This measure, illustrated by Eq. (6), gives the position of a sentence with respect to the borders of a source document [32].

$$\text{LOC\_}B = \max\left(\frac{1}{n}, \frac{1}{s - n + 1}\right) \tag{6}$$

- **COUNT_W**: It is the count of all the words in a sentence [33], as depicted in Eq. (7).

$$\text{COUNT\_}W = w \tag{7}$$

- **COUNT_CH**: The count of the characters that appear in a sentence [33] is given by Eq. (8).

$$\text{COUNT\_CH} = c \tag{8}$$

- **TF**: This measure, denoted by Eq. (9), is obtained by averaging all the "Term frequencies" of every term in a sentence [34].

$$\text{TF} = \frac{\sum_{m \in P} tf(m)}{w} \tag{9}$$

- **TFISF**: It is illustrated by Eq. (10) [35],

$$\text{TFISF} = \sum_{m \in P} tf(m) \times \text{isf}(m) \tag{10}$$

where inverse sentence frequency (ISF), isf(m) is depicted by Eq. (11),

$$\text{isf}(m) = 1 - \frac{\log(s(m))}{\log(s)} \tag{11}$$

where $s(m)$ is the number of sentences that have the term "m" appearing in them.

- **HEADING_O**: It represents the similarity of sentence "*P*" with the document heading "*H*" given by Eq. (12), [30]

$$\operatorname{sim}(P, H) = \frac{|P \cap H|}{\min\{|P|, |H|\}} \tag{12}$$

- **HEADING_J**: It is called as "Jaccard Similarity" calculated for a sentence with respect to the document heading, given by Eq. (13) [30].

$$\operatorname{sim}(P, H) = \frac{|P \cap H|}{|P \cup H|} \tag{13}$$

- **HEADING_C**: Given by Eq. (14) is the "Cosine similarity" of sentence "*P*" with reference to the heading of the document [30].

$$\operatorname{sim}(P, H) = \frac{P \cdot H}{|P||H|} \tag{14}$$

- **T_COMP_O**: Eq. (15) denotes the overlap of the similarity of a sentence "P" with the source document's complement [36].

$$\operatorname{sim}(P, T - P) = \frac{|P \cap (T - P)|}{\min\{|P|, |T - P|\}} \tag{15}$$

- **T_COMP_J**: Eq. (16) represents the "Jaccard Similarity" of a sentence "P" with the complement of the source document. [36]

$$\operatorname{sim}(P, T - P) = \frac{|P \cap (T - P)|}{|P \cup (T - P)|} \tag{16}$$

- **T_COMP_C**: Illustrated by Eq. (17) is the "Cosine Similarity" of sentence "P" computed with reference to the source document's complement [36].

$$\operatorname{sim}(P, T - P) = \frac{P \cdot (T - P)}{|P||T - P|} \tag{17}$$

8 Proposed Methodology

This section describes the supervised ATS methodology used for the summarization of Konkani texts using regression in a step-by-step manner. The steps involved in the process are illustrated using a block diagram, shown in Fig. 1.

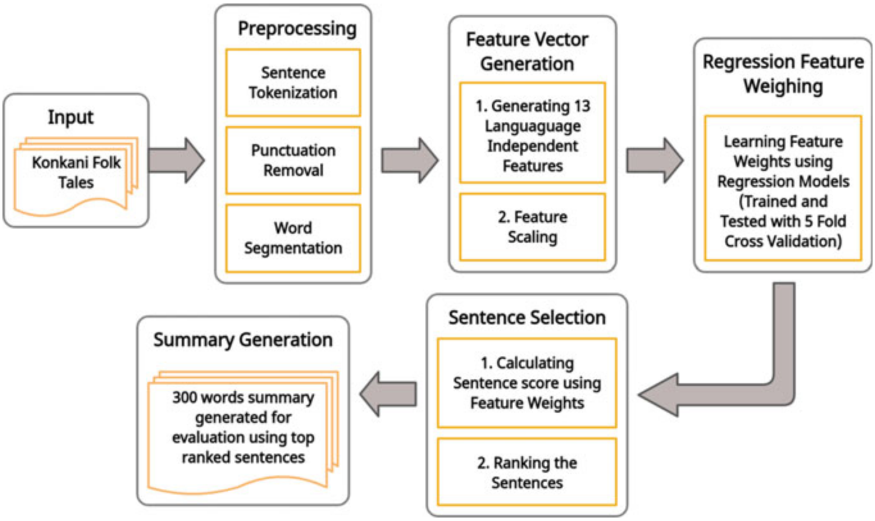


Fig. 1 Supervised ATS methodology block diagram

8.1 Preprocessing

Each sentence is put on a new line. The punctuations are removed from every single sentence from all the documents. These would, however, be preserved in the output summary. The words are separated from the sentences, and then, the cleaned sentences, along with their words, are used for generating feature vectors.

8.2 Feature Vector Generation

Each sentence is translated into a "feature vector." The sentences in each story are depicted as feature vectors together with their corresponding "regressand" that is set to "1.0" when a sentence is to be included in the output summary or set to "0.0" when a sentence is not to be incorporated in the final summary. Each of the features are then scaled between 0 and 1 which brings all the features on a common scale.

8.3 Regression Feature Weighing

The dataset is composed of 71 folk tales and was divided into five folds. In every fold created, 80% of the documents in each fold were used for training and the rest 20% were used for testing. It was not possible to get 5 equal folds as the dataset has 71 stories. Therefore, the first four folds comprised 14 stories and the fifth fold comprised 15 stories for testing. Hence, every document in the dataset gets to be a part of the training set as well as the testing set. During the processing of each fold, the linear regression algorithms were trained to learn the feature weights with the help of the linear regression model. This model estimates the value of a dependent variable when a set of independent variables are supplied to it [12, 21]. The linear regression system receives the dependent and the independent vectors as input; thereafter, a result of a constant weighted value is obtained [12]. Mathematically, the model can be depicted as shown in Eq. (18). The coefficient values learned by the algorithms are used to predict a dependent variable "y." But, we used these coefficient values learnt in the training phase as feature weights to be used in the sentence ranking phase to generate sentence ranking scores.

$$y = W_0 + W_1 x_1 + W_2 x_2 + \cdots + W_n x_n + \varepsilon \tag{18}$$

where

"y" is the dependent variable.

"W_0" is the intercept value.

$W_1, W_2, \ldots, W_n$ are the feature weights or coefficients values.

$x_1, x_2, \ldots, x_n$ are the independent variables.

"ε" is the error term.

8.4 Sentence Selection

During the testing phase of the folds, the sentence scores for each of the sentences are calculated, and then, the sentences are ranked in decreasing order of their scores. Using the feature weights learnt, the sentence scores are computed using Eq. (19).

$$\begin{pmatrix} R_1 \\ R_2 \\ \vdots \\ R_m \end{pmatrix} = \begin{pmatrix} F_{01} & F_{02} & F_{03} & \cdots\cdots & F_n \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ F_{m1} & F_{m2} & F_{m3} & \cdots\cdots & F_{mn} \end{pmatrix} \cdot \begin{pmatrix} W_1 \\ W_2 \\ \vdots \\ W_n \end{pmatrix} \tag{19}$$

In Eq. (19),

"R" represents ranking score of each of the m sentences.

"F" is the feature matrix, where each matrix row represents the features of each sentence in a document present in the testing set.

"W" represents the values of weights learnt from the previous step.

8.5 *Summary Generation*

The sentences of a document are ranked according to their score, and only, the top-ranked sentences are included in the output summary. An upper limit of 300 words is applied to restrict the output summary length. This threshold value is set to 300 words for the summary evaluations since the human-generated summaries are also of 300 words length. Every story in the dataset is summarized by two Konkani language experts. These human-generated summaries are compared using ROUGE toolkit with the system-generated summaries to check for overlap of content.

9 Results and Discussions

We have implemented a system for generating extractive summaries of Konkani folk-tales using supervised ML approaches using linear regression. The dataset comprises 71 Konkani language texts written in Devanagari script and the summarization system produced extractive summaries for each of the documents. The evaluation of these system-generated summaries was accomplished with the help of ROUGE toolkit [37].

ROUGE toolkit functions very well with human estimates and uses N-gram statistics. The system-generated extractive summaries of 300 words and the corresponding pair of abstractive summaries created by human summarizers are compared with one another to gauge the quality of the system's output [33]. The reason is that the output produced by ATS system needs to closely match the "gold-standard" summaries that humans can generate. If the composition of the words in the system-generated summary is comparable to the human-generated "gold-standard," then it signifies a rather fluent output.

ROUGE metric helps in estimating the overlap of "uni-grams," given by ROUGE-1 scores, "bi-grams" given by ROUGE-2, and "Longest Common Subsequence (LCS)" given by ROUGE-L, between the system-generated and human-generated summaries. The specifics of these metrics are described by Lin [37]. ROUGE-1 "uni-gram" scores are given in Table 1. ROUGE-2 "bi-gram" values are shown in Table 2, and ROUGE-L (LCS) points are illustrated in Table 3. The idea of using these metrics is to measure the fluency of the system-generated summaries across different granularities of text. It can be observed from Tables 1 and 2 that ROUGE-1 scores are greater than ROUGE-2 scores; this is because that there could be an identical term in both these ROUGE metrics; however, the converse may not be true. But, ROUGE-L scores are higher than ROUGE-2 scores since ROUGE-L tracks terms appearing in the sentence in order but not necessarily consecutive. Therefore, ROUGE-L integrates longest common n-grams occurring in a sequence.

The performance of the systems implemented based on linear regression in this paper was evaluated against the performance of a k-means clustering-based system with 3 clusters using the same Konkani dataset [26]. It was also compared with

Table 1 ROUGE-1 uni-gram scores

System	ROUGE-1 (uni-gram)		
	Precision	Recall	*F*-Score
Ordinary least squares regression	0.32652	0.32594	0.32621
Ridge regression	0.32598	0.32657	0.32625
LASSO regression	0.32769	0.32774	0.32769
K-means clustering with three clusters	0.31408	0.31373	0.31388
Lead baseline	0.30147	0.30165	0.30154
Modified Luhn's heuristic baseline	0.31421	0.31349	0.31383
Human-annotated benchmark	0.35844	0.35460	0.35608

Table 2 ROUGE-2 bi-gram scores

System	ROUGE-2 (bi-gram)		
	Precision	Recall	*F*-Score
Ordinary least squares regression	0.08450	0.08463	0.08455
Ridge regression	0.08474	0.08489	0.08481
LASSO regression	0.08737	0.08737	0.08736
K-means clustering with three clusters	0.07942	0.07927	0.07934
Lead baseline	0.08097	0.08103	0.08099
Modified Luhn's heuristic baseline	0.07889	0.07876	0.07882
Human-annotated benchmark	0.11088	0.10908	0.10977

Table 3 ROUGE-L LCS scores

System	ROUGE-L (LCS)		
	Precision	Recall	*F*-Score
Ordinary least squares regression	0.31836	0.31892	0.31862
Ridge regression	0.31851	0.31909	0.31878
LASSO regression	0.31943	0.31948	0.31943
K-means clustering with three clusters	0.30680	0.30644	0.30659
Lead baseline	0.29642	0.29659	0.29648
Modified Luhn's heuristic baseline	0.30822	0.30753	0.30786
Human-annotated benchmark	0.35228	0.34847	0.34994

lead baseline and modified Luhn's heuristic baseline. A lead baseline summary is generated with the first 300 words of a text document. It was used in document understanding conference (DUC). It is a simple baseline yet challenging for text summarizers to outperform [38]. Luhn's heuristic is the earliest statistical method for extractive text summarization [39]. The modified Lunh's heurtistic baseline used Luhn's heuristic method, with YAKE selecting 30 relevant keywords [40]. Sentence scores were computed with the square of the number of keywords divided by the window size. "Window size" is the maximum distance between any two keywords [39]. The highest scoring sentences were then selected to generate 300 words summary.

"Precision" and "Recall" recount the overlap between the system-generated output and the human-generated summaries. "Precision" attempts to identify if the content produced by the system summary is relevant. "Recall" helps in identifying if the contents of human-generated summaries have been captured by the system-generated summaries. "F-score" integrates the precision and recall scores into a single score. All the scores are represented as overlap percentage. We also introduce a human-annotated benchmark to compare the systems, which comprises sentence marked by language experts as significant to be a part of a 300-word summary. The system-generated summaries and the human-annotated summaries are compared and evaluated with the "gold-standard" summaries. The values of ROUGE-1, ROUGE-2, and ROUGE-L are between 0 and 1; a value 0 implies no overlap between system-generated summary and human-generated summary, whereas 1 implies a powerful match between the two. The comparative charts showing the precision, recall, and F-score values for ordinary least squares regression, ridge regression, LASSO regression, k-means clustering with three clusters, lead baseline, modified Luhn's heuristic baseline, and human-annotated benchmark are depicted in Figs. 2, 3, and 4, respectively.

We observed that linear regression-based models could beat the baseline systems; however, they could not beat the human-annotated benchmark. The linear regression-based models could select important sentences in the document using language-independent features, making the approach easily extendable to other low-resource languages. We also noted that ridge regression could improve on the performance

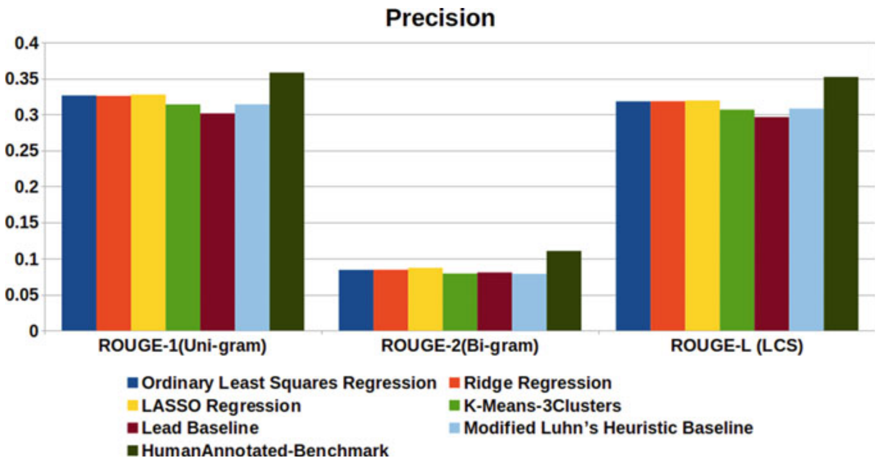


Fig. 2 Comparative chart for precision

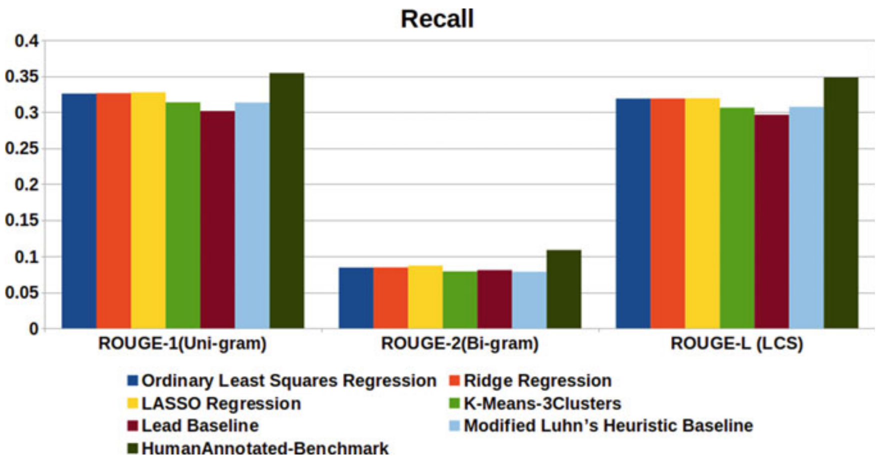


Fig. 3 Comparative chart for recall

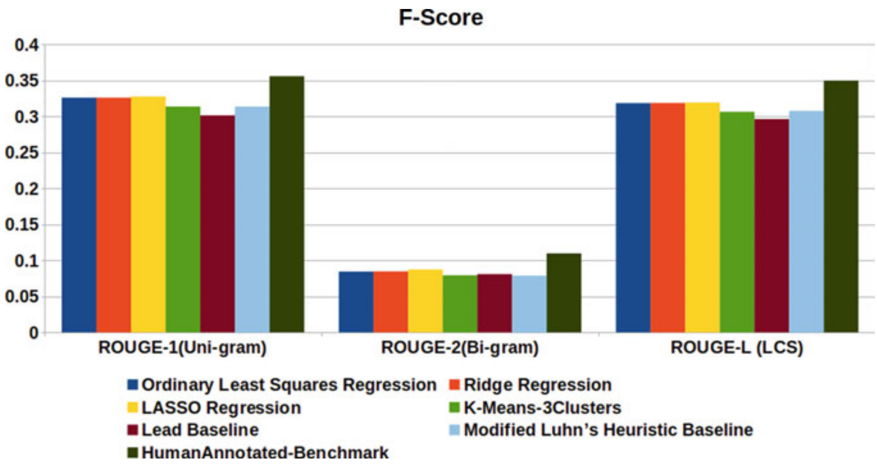


Fig. 4 Comparative chart for F-score

of sentence selection when compared to ordinary least squares. The sentence selection depends on the feature weights learned during the training stage. Unlike ordinary least squares, ridge regression shrinks the values of the feature weights using L2 regularization. This makes the model computationally faster and also reduces the likelihood of over-fitting and allows for a better model fit. LASSO uses L1 regularization which improves on ridge regression as it not only shrinks feature weights but also turns off features that will not be important. This has translated to an even better model fit and, hence, sentence selection. When comparing the systems to the unsupervised ML system, based on k-means method, tested on the same dataset, we see that, being a supervised approach, it can perform better [26].

Using linear regression-based feature weighing along with language-independent features is effective in selecting important sentences from the input document and thus producing an extractive summary.

10 Conclusion

We analyzed the use of supervised linear regression models for the generation of extractive single document summaries in Konkani language. Most of the research in ATS is done on popularly spoken languages in the world, like English. There is a need to extend this research to other less popular languages spoken in the world. Furthermore, the most common genre of the datasets used for such research is news articles, blogs, and reviews. The dataset used in this work belongs to the genre of literature and comprises Konkani language folktales. Folktales have a special significance in the local culture, as the traditions and rituals of a community are passed on to future generations through these stories having a teaching or a moral.

We made use of language-independent features to represent the sentences in the document and used a k-fold evaluation strategy using the ROUGE evaluation toolkit to evaluate the system-generated summaries against the human-generated summaries. The linear regression models are effective at learning feature weights to be used in calculating sentence ranking scores. From our experiments, we observed that our summarization systems, based on linear regression algorithms and language-independent features, could produce acceptable summaries despite not being able to beat our human-annotated extractive benchmark. However, the systems could beat the baselines. In particular, we also observed that using L1 regularization produced better results than L2 regularization and no regularization. Additionally, the language-independent features used can be effortlessly extended to other languages, including low-resource languages, like Konkani, and yet produce promising summaries.

The major limitation of working with a low-resource language, such as Konkani, is the lack of task-specific resources and data. Also, language preprocessing tools such as lemmatizers are not available. Another challenge is the requirement of a human-annotated corpus, for training the ML algorithms, along with human-generated abstracts for evaluation of system-generated summaries.

In future work, the expansion of the size of the dataset can be considered. Other unsupervised approaches, like graph based, semantic based, can be considered for extractive ATS. Also, supervised approaches, including deep learning, can also be explored. Further, we can also consider creating an ensemble of extractive summarizers, which could outperform an individual system.

References

1. Hobson SP, Dorr B, Monz C, Schwartz R (2007) Task-based evaluation of text summarization using relevance prediction. Inf Process Manag 43:1482–1499
2. Ling X, Jiang J, He X, Mei Q, Zhai C, Schatz B (2007) Generating gene summaries from biomedical literature: a study of semi-structured summarization. Inf Process Manage 43(6):1777–1791
3. Lloret E, Palomar M (2012) Text summarisation in progress: a literature review. Artif Intell Rev 37(1):1–41
4. Andhale N, Bewoor LA (2016) An overview of text summarization techniques. In: 2016 International conference on computing communication control and automation (ICCUBEA). IEEE, New York, pp 1–7
5. Moratanch N, Chitrakala S (2017) A survey on extractive text summarization. In: 2017 international conference on computer, communication and signal processing (ICCCSP). IEEE, New York, pp 1–6
6. D'Silva J, Sharma U (2019) Automatic text summarization of Indian languages: a multilingual problem. J Theor Appl Inf Technol 97(1). ISSN: 1992-8645
7. Saleh AA, Weigang L (2017) Language independent text summarization of western European languages using shape coding of text elements. In: 2017 13th International conference on natural computation, fuzzy systems and knowledge discovery (ICNC-FSKD). IEEE, New York, pp 2221–2228
8. D'Silva J, Sharma U (2019) Development of a Konkani language dataset for automatic text summarization and its challenges. Int J Eng Res Technol 12(10). ISSN: 0974-3154. https://doi.org/10.5281/zenodo.5531954
9. Statement—2 (2011) Distribution of population by schedule and other languages India, States and Union Territories—2011. Office of the Registrar General & Census Commissioner, India, Ministry of Home Affairs, Government of India. Available: https://censusindia.gov.in/2011Census/Language-2011/Statement-2.pdf. Accessed 15 Nov 2020
10. Kamat K (1997) The origins of the Konkani language. Konkani Heritage Album. Available: http://www.kamat.com/kalranga/konkani/konkani.htm
11. Ulrich J, Carenini G, Murray G, Ng R (2009) Regression-based summarization of email conversations. In: Proceedings of the international AAAI conference on web and social media, vol 3(1)
12. Hannah ME, Mukherjee S, Kumar KG (2010) An extractive text summarization based on multivariate approach. In: 2010 3rd International conference on advanced computer theory and engineering (ICACTE), vol 3. IEEE, New York, pp V3–157
13. Xie S, Liu Y (2010) Improving supervised learning for meeting summarization using sampling and regression. Comput Speech Lang 24(3):495–514
14. Gupta V, Lehal GS (2011) Features selection and weight learning for Punjabi text summarization. Int J Eng Trends Technol 2(2):45–48
15. Aristoteles A, Widarti W, Wibowo ED (2014) Text feature weighting for summarization of documents Bahasa Indonesia by using binary logistic regression algorithm. Int J Comput Sci Telecommun 5(7):29–33
16. Kumar A, Yadav J, Rani S (2015) Automatic text summarization using regression model (GA). Int J Innov Res Comput Commun Eng 3(5), ISSN (Online): 2320-9801. ISSN (Print): 2320-9798
17. Dlikman A, Last M (2016) Using machine learning methods and linguistic features in single-document extractive summarization. In: Cellier P, Charnois T, Hotho A, Matwin S, Moens M-F, Toussaint Y (eds) Proceedings of DMNLP, workshop at ECML/PKDD, pp 1–8, Riva del Garda, Italy
18. Dlikman A, Litvak M, Last M (2019) Rich feature spaces and regression models in single-document extractive summarization. In: Multilingual text analysis: challenges, models, and approaches, pp 119–154

19. Oliveira H, Lins RD, Lima R, Freitas F, Simske SJ (2017) A regression-based approach using integer linear programming for single-document summarization. In: 2017 IEEE 29th International conference on tools with artificial intelligence (ICTAI). IEEE, New York, pp 270–277
20. Zopf M, Mencía EL, Fürnkranz J (2018) Which scores to predict in sentence regression for text summarization? In: Proceedings of the 2018 Conference of the North American chapter of the association for computational linguistics: human language technologies, vol 1 (Long Papers), pp 1782–1791
21. Khadem S, Ali ZH (2018) Multi-document text summarization based on multiple linear regression. AL-Mansour J 30
22. Rezaei H, Moeinzadeh SA, Shahgholian A, Saraee M (2019) Features in extractive supervised single-document summarization: case of Persian news. arXiv preprint arXiv:1909.02776
23. Cagliero L, Quatra ML (2020) Extracting highlights of scientific articles: a supervised summarization approach. Expert Syst Appl 113659. https://doi.org/10.1016/j.eswa.2020.113659
24. El-Kassas WS, Salama CR, Rafea A, Mohamed HK (2021) Automatic text summarization: a comprehensive survey. Expert Syst Appl 165:113679
25. Gambhir M, Gupta V (2017) Recent automatic text summarization techniques: a survey. Artif Intell Rev 47(1):1–66. https://doi.org/10.1007/s10462-016-9475-9
26. D'Silva J, Sharma U (2020) Unsupervised automatic text summarization of Konkani texts using K-means with Elbow method. Int J Eng Res Technology 13(9):2380–2384. https://dx.doi.org/10.37624/IJERT/13.9.2020.2380-2384
27. D'Silva J, Sharma U (2021) Automatic text summarization of Konkani Folk tales using supervised machine learning algorithms and language independent features. IETE J Res (in Press). https://doi.org/10.1080/03772063.2021.1987993
28. Pedregosa F et al (2011) Scikit-learn: machine learning in Python. J Mach Learn Res 12:2825–2830
29. Litvak M, Last M (2013) Cross-lingual training of summarization systems using annotated corpora in a foreign language. Inf Retrieval 16(5):629–656
30. Edmundson HP (1969) New methods in automatic extracting. J ACM (JACM) 16(2):264–285
31. Baxendale PB (1958) Machine-made index for technical literature—an experiment. IBM J Res Dev 2(4):354–361
32. Lin CY, Hovy E (1997) Identifying topics by position. In: Fifth conference on applied natural language processing, pp 283–290
33. Nobata C, Sekine S, Murata M, Uchimoto K, Utiyama M, Isahara H (2001) Sentence extraction system assembling multiple evidence. In: NTCIR
34. Vanderwende L, Suzuki H, Brockett C, Nenkova A (2007) Beyond SumBasic: task-focused summarization with sentence simplification and lexical expansion. Inf Process Manage 43(6):1606–1618
35. Neto JL, Santos AD, Kaestner CA, Freitas AA (2000) Generating text summaries through the relative importance of topics. In: Advances in artificial intelligence. Springer, Berlin, Heidelberg, pp 300–309
36. Litvak M, Last M, Friedman M (2010) A new approach to improving multilingual summarization using a genetic algorithm. In: Proceedings of the 48th Annual meeting of the association for computational linguistics, pp 927–936
37. Lin CY (2004) Rouge: a package for automatic evaluation of summaries. In: Text summarization branches out, pp 74–81
38. Nenkova A (2005) Automatic text summarization of newswire: lessons learned from the document understanding conference. In: Proceedings of the 20th National conference on artificial intelligence, vol 3, AAAI'05, pp 1436–1441
39. Luhn HP (1958) The automatic creation of literature abstracts. IBM J Res Dev 2(2):159–165
40. Campos R, Mangaravite V, Pasquali A, Jorge A, Nunes C, Jatowt A (2020) YAKE! Keyword extraction from single documents using multiple local features. Inf Sci 509:257–289

Computation Offloading Scheme Classification Using Cloud-Edge Computing for Internet of Vehicles (IoV)

Kumar Gourav and Amanpreet Kaur

Abstract In recent years, there is development in the field of computing devices; Internet of Things (IoT) becomes the latest trend. IoT comprises ubiquitous things that are associated with day-to-day life of individuals like smartphones, smart TV, laptops, and now vehicles too. Internet of Vehicles (IoV) has become the latest area of research used to develop applications in the field of traffic management and road safety. A collaborative approach of cloud and edge computing is termed cloud-edge computing. To manage the enormous amount of IoT devices and the coordination among IoT, the cloud and edge concept of computation offloading is required. In the process of computation offloading, tasks are computationally offloaded to the cloud data center that enhances the resource utilization of the cloud server and minimizes the energy utilization for the tasks. This paper represents the literature review related to various computational offloading schemes in cloud-edge computing proposed as part of the study. The resources comprise of related book chapters and research papers from different publishers of international and national reputes. The study is carried out with the analysis of various computation offloading schemes in cloud-edge computing for the Internet of Vehicles. In addition, computing technologies like cloud computing, edge computing, and computation offloading for the Internet of Vehicles (IoV) were also discussed.

Keywords Internet of Things (IoT) · Edge computing · Cloud computing · Computation offloading · Internet of Vehicles (IoV)

K. Gourav (✉)
University Institute of Computing, Chandigarh University, Chandigarh, India
e-mail: aroragourav28@gmail.com

A. Kaur
Chitkara University Institute of Engineering & Technology, Chitkara University, Rajpura, Punjab, India

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_39

1 Introduction

As there is an increase in the urban population and rapid development of cities over the past few years, the transport segment becomes an important aspect of modern civilization. Due to this rapid development in the field of transportation, vehicles have emerged as an essential commodity in the life of individual well-being [1]. As per the research conducted by World Economic Forum [2], the number of globally registered vehicles is more than 1.5 billion and this figure is projected to be double in the coming years. As there is a tremendous rise in the expansion of vehicles, it is challenging to accomplish with the conventional transportation framework due to this, and the possibility of accidents increases that threatens people's safety. To deal with such a situation, a significant and emergent paradigm of the Internet of Things (IoT) and the Internet of Vehicles (IoV) has developed as a new area for the growth and development in the range of industrial operations in smart and urban cities traffic management system [3]. The development in the field of wireless technologies and IoT puts us accelerative and is dependent upon these smart devices like smartphones and smart vehicles that can implement several powerful and innovative applications Raza et al. [4]. IoV means that vehicles observe the status information inside the vehicle through sensors and actuators and then generate all the dynamic information related to vehicles within the local platform through the infinite communication network and use it effectively Si-Feng et al. [5]. The Internet of Vehicles not only transmits communication wirelessly between vehicles, road figures, and data related to pedestrians but also intelligently recognizes traffic control and vehicles control and manages network and lively information service [6]. A huge amount of data is generated by this novel technology.

Cloud computing is used to manage such situations by providing on-demand resources features to the users. Clouds are also used to store data, process data, and analyze data for IoV. During the last few years, this cloud computing becomes the fastest developed technology used for consistent data centers that are spread on huge areas and also associated to make reliable services [7]. Meanwhile, some applications related to vehicular networking have been positioned to the cloud to offer related services to the users. The number of vehicles and mobile terminals increases rapidly that causing the load to the cloud and making it heavier. Apart from cloud computing, data centers are also comparatively far from the end-users causing high processing latency and becoming a serious issue for the latency-sensitive applications in the IoV. This issue of cloud computing has led edge computing to come into existence Tang et al. [8]. Data is usually created at the edge of the network. So it is reasonable to process the data nearer to the location instead of moving it far away to the cloud. By using Edge Computing, end-users preprocess their tasks by offloading them for implementation to the edge computing devices (ECDs). While computing the tasks using edge computing, resources that are nearer to the terminal cause a huge number of offloading requests. Edge computing should not alone replace cloud computing but both these computing technologies should complement each other to encounter the offloading requests and makes the computing tasks better.

Computation offloading is the technology used to assign a considerable amount of computation to a computation node with adequate resources for treating and recovering the computing results from the computation node. Edge computing is considered one of the critical technologies; computation offloading is mainly distributed into resource allocation and offloading decisions. According to Li et al. [9] paid emphasis on when to offload tasks, how many tasks will offload, and where to offload tasks from the viewpoint of the offloading decision problem, and decision problem is accomplished when transmission delay and energy utilization for terminals are caused by offloading the tasks.

This paper reviews the computing technologies with computation offloading strategies, processing steps, and offloading techniques which were also discussed. The study also reviews the different computation offloading schemes proposed by different researchers as part of their study for the Internet of Vehicles (IoV). The rest of this paper is organized as follows: Sect. 2 discussed the background of computing technologies, computation offloading, and the Internet of Vehicles. Section 3 discusses the related work and literature review about various computation offloading schemes. At last, Sect. 4 concludes the study.

2 Literature Review

With the expansion of vehicles increasing over the previous years, problems related to traffic arise frequently [10]. Further, the Internet of Vehicles seems to become the solution to these rapidly increased problems. As IoV has gained momentum these days, various changes get experienced in this field. Zhang [11] explained in his study that the growth in the field of wireless technologies and its applications and interconnected smart vehicles are considered as the future trend and become an automotive revolution. Several related works are to be carried out in vehicular cloud networks. In the study of Zhang et al. [12], the Internet of Vehicles is considered as a complex system as it demands feedback and real-time information handling but conventional cloud platforms cannot be appropriate for these situations. Gao et al. [13] in their study stated that cloud platforms cannot encounter the prerequisite for computing knowledge and low latencies. Therefore, Wang [14] used edge computing (EC) and Cheang [15] used mobile edge computing (MEC) technologies that are the appropriate solution to the environment of the Internet of Vehicles and growing rapidly. This technology can provide stable transmission and a faster data processing rate. It also can offer additional storage for vehicles and computing resources to the end-users when compared with traditional cloud computing [16–18]. Wang et al. [19] used mobile edge computing in IoV because it minimizes the distance between the vehicle and the server and offers effective data processing. According to researcher [20], computation offloading is used to free the complex loads from the servers by using available resources. In mobile edge computing, offloading was proposed to shift the heavy jobs to edge computing devices (ECDs) for implementing the results with minimum energy consumption and minimum latencies and also improves the quality of computing knowledge. Author [21] examined in his study that the devices used in edge computing for computation offloading are part of energy utilization.

Number of studies have been conducted in the field of multi-objective optimization for offloading computing the responsibilities in edge computing on Internet of Vehicle. It is still a problem and yet to be resolved by taking into account energy utilization and time in load balancing. To address these challenges, many offloading methods and algorithms are proposed by many researchers.

2.1 State of the Art-Related Research

In this section, a set of several computation offloading papers related to the Internet of Vehicles that are published in various reputed journals in the period between 2017 and 2021 will be reviewed below.

Xu et al. [22] suggested a method for computational offloading empowered with cloud-edge computing for the Internet of Vehicles termed as multi-objective computation offloading (MCO) method. Researchers considered cloud-edge computing for the Internet of Vehicles due to the limited and inadequate number of resources in the edge computing devices that are unable to deal with the huge amount of data. To process these tasks, cloud was used. Hence, a hybrid computing concept was used in the Internet of Vehicles that increases the well-being of traffic and accessibility for the users but it is unsatisfactory while managing high energy consumption and extended execution time. To overcome these problems, the author proposes MCO method that stresses enhancing the load balancing rate, execution time for tasks cloud-edge computing process, and energy utilization of edge computing devices making the information reliable on the Internet of Vehicles. In related work, the researcher carries out in his study that for the better allocation and coordination of resources among edge servers and cloud servers that improved performance of tasks in edge computing devices, computation offloading methods are proposed.

Further in his study, the researcher designed a route locating algorithm for the vehicle-to-vehicle transmission to get the path of the vehicle from the origin, i.e., from where the task was originated to the goal vehicle where the tasks get offloaded. To accomplish the multi-objective development of decreasing the time for computation of the tasks and utilization of energy when load balancing was taken into consideration, Non-dominated Sorting Genetic Algorithm III (NSGA-III) was implemented. To estimate the efficiency of the purposed MCO, the author employed Multiple Criteria Decision-Making (MCDM) and Simple Additive Weighting (SAW) and some experimental evaluations were implemented by using CloudSim simulator.

Researcher [23] concentrated on the saving of energy in edge computing and framed an algorithm to enhance the communication control of applications offloading among Road Side Units and vehicles.

Dai et al. [24] projected a combined problem based on balancing load and offloading to escalate the efficacy of the vehicular edge computing system. To design it perfectly, both transmission time and the computation time were taken into consideration. The primary objective to formulate this problem was to deal with the challenges like selecting the accurate vehicular edge computing server for balancing the load and offloading and obtaining the best offloading solution to enhance the system performance. These challenges are overcome by developing the joint optimal VEC server selection and offloading (JSCO) algorithm with minimum complexity that mutually enhances decision of selection, ratio offloading, and computation of resources. To authenticate the efficiency of JSCO benchmark schemes like Selection Optimization scheme (SO), Computation Offloading scheme (CO) and The Brute Force Scheme (BFS) was used. It depicts the performance of the algorithm suggested that perfectly works for balancing the load and enhances the performance of the system.

Wan et al. [25] proposed a framework for edge computing to offload the Internet of Vehicles named COV. To select the appropriate destination edge nodes, problem by using multi-objective optimization technique was formulated particularly to lessen the offloading interruption, the cost for offloading, and enhancing the load balance at a similar period. Strength Pareto Evolutionary Algorithm 2 (SPEA2) was selected to resolve the above-mentioned problem and to choose the desired goal edge nodes because of its parallel processing mechanism, high performance, simple and universal, good robustness, and global optimization in multi-objective development problems. To find the suitable and efficient computation offloading strategy, the author adopted Technique for Order Preference by Similarity to an Ideal Solution (TOPSIS) and Multi-Criteria Decision-Making (MCDM). To validate the efficacy and efficiency, some comparative methods were implemented by the CloudSim framework.

Xu et al. [26] proposed ECO as an edge computing computation offloading technique to examine the privacy conflicts related to the computing responsibilities that are transferred to the edge computing devices and computation offloading problem for IoV. Vehicle-to-vehicle (V2V) communication was designed that was based on a pathfinding algorithm to obtain the path from where the task was originated, i.e., source vehicle to where the task was offloaded, i.e., destination vehicle. To achieve multi-objective optimization, to minimize the task accomplishment time, to curtail the energy utilization of edge computing devices, and to guard against privacy conflicts for computing tasks, Non-dominated Sorting Genetic Algorithm II (NSGA-II) was utilized. Fitness function was used to evaluate the outcomes generated by genetic algorithms. This fitness function contains minimum task accomplish time and minimum energy utilization that displays the conduct of the proposed method and then balances the load for the resources. Consecutive preliminary assessments authenticate the capability and validity of ECO, and some comparative methods were implemented by the CloudSim framework.

Zhang [27] projected a mobile edge cloud-based framework for vehicular networks related to computational offloading that shows the computational efficiency of the transfer structures of vehicle-to-interface and vehicle-to-vehicle methods of communication. Furthermore, a relegation scheme on, effective predictive combination mode was proposed that considered the mobility of the vehicles and time consumption of the execution of responsibilities.

Xu et al. [28] suggested a method for computation offloading in his study named Vehicle to Everything Communication (V2X-COM) that was based on V2X communication in edge computing for finding offloading routes related to computing tasks that are offloaded among the goal vehicle and the destination vehicle. To create balanced offloading strategies like developing resource implementation of edge computing devices (ECDs) and reducing the latency, Non-dominated Sorting Genetic Algorithm (NSGA-III) was used. To verify the efficiency of V2X-COM and to discover the ideal offloading approach, Simple Additive Weighting (SAW) and Multiple Criteria Decision-Making (MCDM) were executed with some comparative methods by using the CloudSim simulator.

Zhao et al. [29] formulated a combined offloading problem for vehicular networks that is dependent on the commonly used technologies cloud computing and mobile edge computing. The formulated problem escalates the structure efficacy by improving the computation resource allocation and computation offloading decisions mutually. Collaborative Computation Offloading and Resource Allocation Optimization (CCORAO) pattern was recommended to divide the issue into two sub-issues of resource allocation that was completed by utilizing the Lagrange multiplier method and computation offloading decision-making by taking the offloading results by using the game-theoretic approach. Distributed Computation Offloading and Resource Allocation algorithm (DCORA) was also developed that can reduce the structural complexity without affecting its execution. The results generated by the simulator depict that the suggested algorithm can efficiently improve the system efficacy and computation time, particularly for the situation in which the mobile edge computing servers could not satisfy the requirements because of inadequate computation basics. The study also reveals that the computing resources of mobile edge computing servers decrease as there is an increase in the amount of computation-intensive responsibilities or the expenditure for the computation resources.

Liwang et al. [30] projected an architecture by combining the cloud-enabled Internet of Vehicles with a satellite network that offers resource management and seamless coverage. A problem using incentive mechanism for resourceful computation offloading in which vehicles were displayed under cost and delay constraints. To find optimal solutions, Markov Chain Monte Carlo (MCMC) and Metropolis criterion algorithms were used that were based on simulated annealing heuristic algorithm.

In 2019, Li [31] shows in his study a computational offloading approach used in self-organizing vehicle networks and elevates the traditional computing offloading approach. The researcher proposed a partial flooding algorithm that was based on a vehicle computing environment but there was no infrastructure associated with it but it was the complete evaluation method for computing power and system reliability. For better performance of partial flooding algorithm, nodes were selected in the vehicle-to-vehicle communication that can moderate task accomplish interval, boost the reliability of system user involvement, and also reduce the influence of vehicle flexibility on offloading. The findings of the study show that the partial flooding algorithm not only improves the offloading conduct of the entire system but also improves the utilization of computing resources.

According to Lin in 2019 [32], computation offloading was related to edge computing and becomes distributed computing model. This computing model was the significant trend of computation offloading to edge computing that will distribute network traffic and reduce latency. Further, the study also reviews the features of edge computing with challenges related to offloading estimation related to task allocation, segregation of applications, distributed execution, and managing resources, and recent efforts are examined to deal with these challenges and elucidate their solutions by exploring some latent technologies like blockchain, serverless computing, NFaaS, and combined domain-specific programming models. The researcher illustrated some distinctive application scenarios like vehicle communication, smart things, cloud gaming, etc. that visualizes the benefits of edge computing and computation offloading.

In 2020, Hou [33] proposed an architecture named Edge Computing-enabled Software-Defined Internet of Vehicles (EC-SDIoV) that provides low latency computing services through Software-defined Networking (SDN) by using nodes of mobile edge computing and nodes of static edge computing. In this study, the researcher focused on the reliability of computation offloading by considering the task allocation partial offloading and reprocessing mechanism mutually. Fault-tolerant Particle Swarm Optimization algorithm (FPSO-MR) a heuristic algorithm is intended for boosting the consistency latency resources. To authenticate the proposed architecture, simulation outcomes confirmed that the model is proficient for decreasing the latency and improving the consistency.

Xu et al. [34] developed an offloading method titled ACOM (Adaptive Computation Offloading Method) for edge computing in the Internet of Connected Vehicles (IoCV) to decrease the transmission interval and to increase the utilization of resources. There is a joint utilization of Macro Base Stations (MABSs) and Road Side Units (RSUs) in IoCV for managing the vehicular tasks. To generate available strategies algorithm based on decomposition, Multi-Objective Evolutionary (MOEA/D) was selected, and by using the utility valuation techniques, optimal strategy was also obtained. Further, the comparative analysis and experimental evaluation results validate the ACOM.

Researchers in 2020 [35] construct the model of task offloading for multi-server, multi-user vehicle edge computing situations and also define the offloading efficacy of the vehicles. Further, the researcher formulates an assimilated problem based on integer nonlinear programming combined with the task scheduling and offloading decision problems to maximize system offloading effectiveness. To generate the ideal solution and to minimize the time complexity, author projected a compound optimization algorithm named Joint Offloading Decision and Task Scheduling Algorithm (JODTS) that is the combination of Partheno Genetic Algorithm (PGA) and a set of experiential rules. Various simulation outcomes show that the projected algorithm efficiently improves the offloading utility of the vehicle edge computing situations and is appropriate for task offloading in different situations.

Wang et al. [36] highlighted the association among multiple mobile edge computing servers and proposed a smart management system for vehicles that offers mobility-aware, computation-intensive, and minimum latency services. To decrease the delay and to explain the optimization issue related to the allocation of resources in various kinds of Internet of Vehicles tasks, researcher formulated the problem for optimization as Markov decision in which the space for state, action, and reward function was defined. Further, he designed a smart and extensible resource distribution algorithm through deep Q-learning to process the data with high dimensions. Simulation outcomes demonstrate that the suggested algorithm can attain improved performance related to time efficiency and load balance and interrupt the prospect that specifies the possibility and effectivity of the proposed system.

In 2020, Chen [37] suggested a scheme for task offloading based on vehicle-to-vehicle (V2V) transmission by analyzing the various useless properties of assembled vehicles and decreasing the processing time. Further, the researcher formulated the execution of tasks as a Min–Max problem where a single task and many other supportive offloading patterns were used to optimize the total task execution time. To determine the finest task allocation system and to reduce the processing time, Max–Min Fairness scheme along with Particle Swarm Optimization (PSO) algorithm is utilized. To generate vehicle track files, IDM_IM model served by VanetMobiSim is to be adopted. Simulation outputs determine the efficiency of the suggested scheme.

Wang et al. [38] recommended the online offloading scheduling and resource allocation (OOSRA) procedure that increases the real-time presentation of vehicle edge computing systems. Further, the study contributes in three ways. Firstly, an online computation offloading scheduling algorithm is modeled that is dependent on the dynamic Cournot game model that is online and requires a small amount of communication overhead. Secondly, an online resource allocation algorithm is to be designed by using the first fit algorithm to decrease the total time and to reduce energy utilization. To validate the proposed algorithm Veins framework that is the combination of network and traffic simulator (Simulation of Urban Mobility (SUMO) considered as simulator for traffic and OMNetCC considered as network simulator) and also extended as the simulation engine for Vehicle Edge Computing (VEC) system.

Ning et al. [39] presented a scheme MEC-enabled Energy-Efficient Scheduling (MEES) to satisfy latency constraints of tasks and to reduce the energy utilization for MEC-enabled roadside units. It also includes energy consumption estimation, processes of delay estimation, result feeding back, and task scheduling and processing. Monte Carlo simulations were used to generate the input randomly and then obtain the statistical value as output. To validate the efficiency of MEES, All Task Admission Algorithm (ATAA) and GMCF were considered. The findings of this study show its efficiency related to latency, energy efficiency, and task blocking possibility. It also shows the aspects of high-efficient offloading solutions and energy management of RSUs. Moreover, the researcher also considered some research challenges and problems related to the characteristics for managing the energy resources of Road Side Units and adequate scheduled offloading.

Garg in 2019 [40] proposed a framework with smart security for Vehicular Ad hoc Networks (VANETs) to intensify the abilities of transmission and computation which was the combination of edge computing nodes and computation technology. V2V and V2E scenarios of transmission in VANETs are used to identify the outbreaks that occur during concurrent information produced by the system. This was completed by using the probabilistic data structure approach quotient filter. To simulate the proposed framework, MATLAB is used where the three-layer arrangement was considered in which the first layer specifies the quantity of nodes in the system, the second layer contains information related to edge nodes, and the last layer is expressed as the cloud storage and the ultimate center for data. The findings of the study were to outperform the vehicular model which provides the energy-efficient secure structure with the least interruption.

Raza et al. [4] in his article discussed the architecture of vehicular edge computing (VEC). This architecture was developed to maintain the real-time information distribution and high level of scalability and mobility. VEC model was suitable for the vehicles that reduce latency for facilities that were capable of taking decisions for real-time situations. This proposed model improves the performance when related with the conventional system by supporting intelligent vehicular computing and producing the optimal utilization of the available alternatives. Furthermore, the researcher also considered all the technical problems related to VEC architecture and go through with its appropriate and latest solutions.

In 2021, Li et al. [9] proposed a computation offloading approach by using cloud-edge computing technology collectively to encounter the offloading specifications and to maximize resources utilization. An algorithm constructed on semidefinite programming was also proposed to resolve the optimization problem. This proposed algorithm offloads the tasks to the edge server that is adjacent to the base station, then to the neighboring sever, and at last to the cloud center. By performing a set of experiments, the algorithm significantly improves the turnaround time, maximizes the server utilization, and reduces energy utilization.

The study by Long et al. in 2020 [67] proposed a scheme based on computational offloading over mobile vehicles termed COTV in IoT-edge-cloud network. This scheme was used to reduce the energy utilization and delay in communication among the sensing gadgets. Three-layered system architecture was projected for COTV scheme that consists of device layer/vehicle layer, mobile edge computing layer, and center cloud layer. This said architecture processes the collection, transmission, and execution of tasks. Energy utilization and latency were the offloading metrics that were treated in the proposed COTV scheme. To make verdicts, platform based on the technique of deep reinforcement learning was used. By performing a considerable set of experiments, COTV successively decreases the delay and improves the system performance.

Table 1 lists the various proposed computation offloading schemes in the literature using cloud-edge computing for the Internet of Vehicles (IoV).

Table 1 Computation offloading schemes using cloud-edge computing for IoV

References	Methodology/technique used	Parameters used	Findings
[22]	Multi-objective computation offloading method (MOC)	V2V Communication, NSGA-III, SAW, MDCM and CloudSim simulator	Cloud-edge computing-enabled computation offloading method is proposed that highlights on improving load balancing rate, the execution time for tasks in IoV-CEC, and energy consumption of ECDs
[24]	Joint optimal VEC Server Selection and Offloading (JSCO) Algorithm	Vehicular edge computing	JSCO algorithm is proposed to deal with challenges like selecting the edge server for balancing the load and offloading and obtaining the best offloading solution to increase the performance of the system. Partial offloading was preferred over binary offloading
[25]	Computation Offloading method for IoV (COV)	SPEA2, TOPSIS, MCDM, and CloudSim simulator	The multi-objective optimization problem is used to select the appropriate destination edge node through which the offloading strategies are to be formulated that reduces the vehicle offloading delay, offloading cost, and load balance of edge nodes
[26]	Edge computing computation offloading (ECO)	V2V, NSGA-II, and CloudSim simulator	The multi-objective optimization technique is used for offloading computing to reduce the implementation time and energy consumption for edge computing devices that also satisfy the security constraint of the computing tasks and load balancing for the resources

(continued)

Table 1 (continued)

References	Methodology/technique used	Parameters used	Findings
[28]	Vehicle to Everything Computation Offloading Method (V2X-COM)	V2X, NSGA-III, SAW, MCDM, and CloudSim simulator	The computation offloading method is used to reduce the latency and improve resource utilization in the Internet of Vehicles
[29]	Collaborative Computation Offloading and Resource Allocation Optimization (CCORAO)	Lagrange multiplier method, game-theoretic approach, and DCORA	The proposed algorithm can efficiently improve the system utility and computation time, particularly for the situation where the mobile edge computing servers were unable to meet demands due to inadequate computation resources
[30]	Cloud-enabled framework with satellite network for Internet of Vehicles	Markov Chain Monte Carlo (MCMC), Metropolis criterion algorithm and simulated annealing algorithm	A computation offloading model for vehicles is developed to deal with delay and cost constraints where the end-users can either considerably decrease the application completion time by offloading or control monetary costs
[31]	Computational Offloading approach used for self-organizing vehicle network	V2V, partial flooding algorithm	The proposed algorithm successfully progresses the offloading performance of the whole system by completing the delay-sensitive tasks, improving the utilization of resources, and balancing the performance among available systems and resources

(continued)

Table 1 (continued)

References	Methodology/technique used	Parameters used	Findings
[32]	Review article	Edge computing	Edge computing can be treated as a distributed computing model related to computation offloading. It also studies the attributes of edge computing with challenges related to computation offloading, and recent efforts are examined to deal with these challenges and elucidate their solutions
[33]	Edge Computing-enabled Software-Defined Internet of Vehicles (EC-SDIoV)	SDN, FPSO-MR	Computation offloading strategy collectively with the task allocation, partial offloading, and reprocessing mechanism is used to improve the reliability of latency-sensitive applications
[34]	Adaptive Computation Offloading Method (ACOM)	MABSs, RSUs, and MOEA/D	The goal of the proposed model is to decrease the transmission delay and increase resource utilization
[35]	Joint Offloading Decision and Task Scheduling Algorithm (JODTS)	PGA (Partheno Genetic Algorithm) and heuristic rules	Vehicle edge computing is considered where the numerous adjacent mobile edge computing servers provide computation offloading services for passing vehicles, and every server serves multiple vehicles at a time. Task offloading algorithm is used for making decisions and executing the tasks
[36]	Intelligent management vehicular system	Markov decision, flexible resource allocation algorithm through deep Q-learning	There is a high rate of latency, maximum execution time, and high expenses by using the infrastructure or RSUs

(continued)

Table 1 (continued)

References	Methodology/technique used	Parameters used	Findings
[37]	Task offloading scheme based on V2V communication and Min–Max problem	Max–Min Fairness scheme, PSO Algorithm, IDM_IM model and VanetMobiSim	A task scheduling scheme is used where one task vehicle and several service vehicles mutually implement the vehicle-carried tasks. The task vehicle collectively reviews the computing control and enhances the service time for all vehicles and reduces the task implementation time
[38]	Online offloading scheduling and resource allocation algorithm (OOSRA)	Cournot game model, first fit algorithm, Veins framework, SUMO, and OMNetCC	Three-layered vehicle edge computing architecture using online offloading scheduling and resource allocation algorithm is implemented that progresses the performance of the system and increases the real-time performance of vehicle edge computing system
[39]	MEC-enabled Energy-Efficient Scheduling (MEES)	Monte Carlo simulations, ATAA and GMCF	The proposed scheme is used to satisfy the latency constraints of tasks and to minimize the energy consumption and task blocking possibility for MEC-enabled RSUs (Road Side Units). It also includes energy consumption estimation, processes of delay estimation, result feeding back, and task scheduling and processing
[40]	Smart security framework for VANETs (Vehicular Ad hoc Networks)	V2V, V2E, quotient filter, and MATLAB	The research shows the vehicular model that provides the energy-efficient secure system with minimum delay

(continued)

Table 1 (continued)

References	Methodology/technique used	Parameters used	Findings
[4]	Survey Article	Vehicular edge computing (VEC)	VEC model was suitable for vehicles that reduce latency for services that were capable of real-time decision-making. It also improves the computational performance of the system by supporting smart vehicular computing and by making the optimal utilization of the available alternatives
[67]	Computational offloading over mobile vehicles in IoT-edge-cloud network (COTV)	Deep reinforcement learning	Energy utilization and latency were the offloading metrics that were treated in the proposed scheme. COTV successively decreases the delay and improves the system performance
[9]	Computation offloading approach using cloud-edge network	Semidefinite programming	This proposed algorithm offloads the tasks to the edge server, improves the turnaround time, maximizes the server utilization, and reduces the energy utilization

3 Background

3.1 Cloud Computing

Cloud computing is a flourishing computing system in which multiple devices are accessible that can be offloaded and accumulated at cloud data centers. It is the collaboration of various consistent systems that work together to form a single integrated computing resource Arunarani et al. [41]. It is service-oriented computing that offers various kinds of services. This computing technology provides multiple benefits to the end-users like flexibility, mobility of data, prevention of data, security, and software update at very nominal cost. The cloud computing system is divided into two parts: Front-end is the client who accesses the services, and the back-end is the cloud server from where the services are accessed. Both of these ends are connected through the Internet. There is an on-demand facility providing access to the resources only to the authorized cloud service providers. These days cloud services are extensively used to serve clients, and this service is generally provided by cloud administrators or cloud service providers by giving access to a pool of collective services that are accessible on-demand [42]. This invention is responsible of providing many benefits like minimizing time consumption, reducing the cost, and storage capacity. Most of the applications are working on the virtual platforms, and these resources are distributed among these virtual machines Arunarani et al. [41].

Cloud services are classified into three categories, that is Software as a Service (SaaS) in which the facility provider makes available to access various software applications on the Internet, Platform as a Service (PaaS) in which the platform is provided that helps the users to access the services without any installation of software, and Infrastructure as a Service (IaaS) in which the physical resources are being shared by virtualization technology for implementing the services for end-users. There are four types of clouds available each of which is liable on the nature of its usage. Cloud types include private cloud that is used by a single individual and intended with extraordinary privacy and high configurability. Community clouds are used by the set of individuals that have infrastructure which is shared between multiple groups. Public clouds are the generic model of clouds where the set of services are provided by the cloud service providers. These clouds are very economical, easy to handle, and become more common when compared to private clouds. Google is the most prominent example of a public cloud. Lastly, a hybrid cloud is a combination of either of three types of clouds Yousefpour et al. [43]. The tremendous growth in the IoT devices and an enormous quantity of information created at the edge of the network cause low bandwidth, latency, and inefficient resources. The cloud computing model was unable to handle all these issues.

3.2 Edge Computing

Due to the lack of availability of permanent resources by the cloud platforms, there are certain issues to deal with such a huge amount of data that causes transmission delay and uncertainty of connection in various domains; IoV is one of them. To deal with such issues, edge computing becomes the new paradigm. Edge computing is the distributed computing technology that deals with the applications and also with the data facilities at the edge of the network. It is also known as the advanced version of cloud computing Li et al. [44]. Researchers [45] explain in their study that this technology is combined with IoT and then enhances the computation capability and storage of data stored on networks by providing low latency and ongoing access to the system. In edge computing, all the data on the edge server is processed and set at the edge of the Internet. Shi [46] defines that edge computing is the technology that permits computation to be accomplished at the edge of the system. Edge is characterized as the resource path among cloud data centers and the source of the data. To use the resources of cloud computation and storage of information on the cloud nearer to the devices on the network edge and users is the major concern of edge computing. Researchers [47] highlight in their study that the edge computing model is used to reduce the latency between cloud and devices and getting services closer to the users allows accessing the resources of the cloud efficiently. So that the devices express as the data producers as well as data consumers in edge computing technology. In this technology, data that is processed at the edge of the system is referred to as the part of the core system where the linked nodes produce data directly. These nodes are equipped with the various platforms of edge computing that consists of storage, system information, computation, and additional functions that can offload the load of the network. There are various schemes like mobile edge computing (MEC), cloudlets, and fog computing that are available [46] and implemented under the principle of edge computing but they all work on different mechanisms [48]. In brief, edge computing is defined as the figuring model that permits processing and storage of information at the edge and decentralization of intelligent facilities of the cloud at the source of data. It also gathers and processes the data but does not emphasize the storing of data, whereas cloud computing emphasizes the data storage on the cloud servers.

3.2.1 Architecture of Edge Computing

As studied, the servers of edge computing are nearer to the end-users than the cloud server but still, it provides better service to the end-users. The architecture of edge computing as shown in Fig. 1 comprises three layers/parts: front-end, i.e., edge device layer; near-end, i.e., edge server layer; and far-end, i.e., cloud server layer. In edge device layer, end devices can interact and operate physically to accomplish tasks

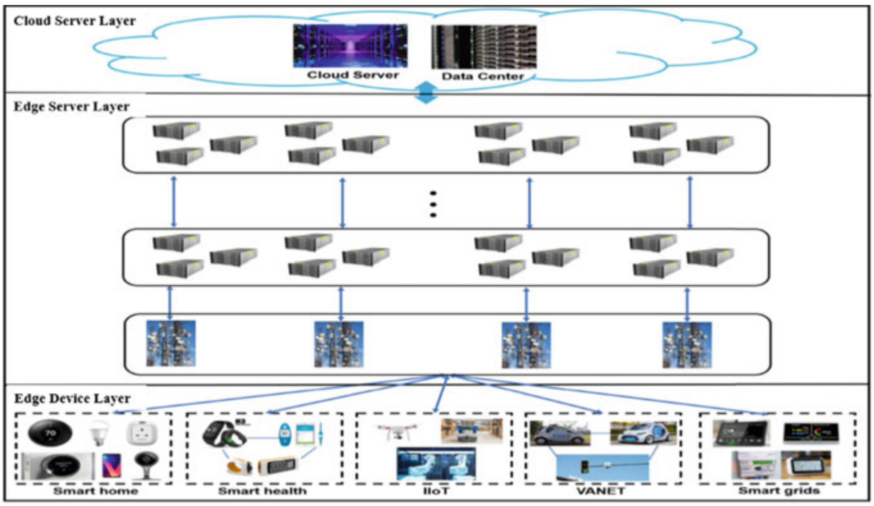


Fig. 1 Architecture of edge computing

like controlling, sensing, and actuating for the users. These devices are organized by microcontrollers with low-level software programmed in them and also with device hardware that is known as firmware that includes the coding of tasks. Here, the devices transferred the requirements to the servers. Edge server layer consists of several sublayers of different edge servers. The server that is at the lowest level has access points and wireless base stations that are used to receive the data from edge devices and send them back control flows by using various wireless interfaces. Then, the data is transferred to the upper sublayers by the base stations or the access points for the computation of tasks. These servers deal with the core computing functions like computation, task offloading, data analytics, authorization, authentication, and storage of data for edge computing. In cloud server layer, cloud servers are used for data storage. These servers and data centers contain a group of powerful machines. The cloud servers are liable for authentication and authorization, integration and computation of various tasks offloaded from edge servers, and data centers for loading of the huge amount of data produced by edge devices and servers.

3.3 Cloud-Edge Computing

Cloud-edge computing is the hybrid prototype of computing in which the development in the services and resources related to the Internet of Things combines the framework of cloud and edge computing. Like cloud computing, edge computing also has small data centers that are known as cloudlets that have an adequate number of access points, resources, and base stations at the edge of the network. These small data centers link the devices through LAN, and the end-users can transfer their information-related responsibilities to edge computing devices to get high bandwidth and minimum latency. In edge computing, users can easily access the devices because of the extensive distribution of these devices. But these edge computing devices have limited resources that may be inadequate to process the enormous flow of data. To deal with such problems, clouds are used to compute the tasks that have huge resource requirements Wang et al. [19].

Hence, the concept of cloud-edge computing came into existence. This framework is the combination of cloud data centers that is also known as cloud plane and edge micro-data centers that is also known as edge plane Jiang et al. [49]. To process the long-term or large-scale global data, cloud planes are used, and to process the short-term or small-scale local data, edge planes are used [50]. Both the cloud data center and edge micro-data centers are the combination of physical devices and a certain number of virtual machines to process the information that is collected by edge devices. By sending some or all of the information nearer to the end-user or the data collection point, cloud-edge computing can alleviate the properties of extensively distributed locations by reducing the effect of latency on the applications. The cloud-edge computing model can fulfill the necessities for the implementation of tasks in specified intervals and utilization of energy resources for the offloaded devices. It also permits the users to transfer their assignments to edge computing devices for better implementation.

3.4 Computation Offloading

Computation offloading is the technique to execute tasks on the remote server that can reduce its execution time and energy utilization that improve the performance of the system [51]. It divides the high energy utilization IoT applications and channelizes the tasks to make efficient use of cloud resources. Applications are divided according to their assistance from resource expansion in three groups: communication-intensive group in which the applications are considered by cloudlets that implements a service model as the network, data-intensive group augments the applications generated by the communication-intensive group, and computation-intensive group augments the applications by using computational offloading [52]. Computation offloading is the significant pattern in cloud-edge computing and rapidly grown as computing power over the systems. The process of computation offloading is to influence the remote servers to enhance the computing ability for smart devices. This model has progressed in the previous years with the growth of different computing skills. Firstly, it was used in mobile cloud computing but now it is integrated with edge computing and mobile edge computing. In this technique, the device transfers a certain amount of data from its computing environment to the server. This offloading procedure includes offloading decisions, dividing the applications, and shared task execution. The computation offloading method also provides requisite computing resources along with various tasks for IoV. The computation offloading mechanism is a significant aspect of the field of vehicular edge computing. By using this technique, the vehicles select their optimal node that satisfies low cost, low latency, and high energy efficiency. The multi-objective computation offloading (MCO) approach is generally used as part of computational offloading for complex tasks that consist of time-constrained functions and simultaneous assignments.

Many researchers used the multi-objective computation offloading method in their study for various purposes. Some utilized this method for balancing load rate, to decrease the time for execution and minimize the energy utilization. Many researchers proposed computation offloading and multi-objective computational offloading with computational offloading algorithms in their study to use the various aspects of this mechanism. Tseng et al. [53] concentrated on the best way to decrease the offloading and implementation time and to utilize the resources provided by cloud-edge computing. The researcher suggested optimization procedures that understand the multi-objective optimization for efficient energy utilization and time utilization. Peng et al. [54] discussed in their study the optimization of time and energy utilization for workflow operations in mobile cloud computing. Further, the researcher proposed a method for multi-objective optimization that is based on the whale optimization method. Xu et al. [55] proposed the information security and offloading utility problem. Further, the researcher designed a multi-objective offloading technique by using Non-dominated Sorting Genetic Algorithm III (NSGA-III) and apprehended the optimization for time utilization of time and resources in mobile edge computing. To compare the results produced by the various proposed multi objective computation offloading methods, comparative methods like first fit and next fit were used. There are various offloading strategies, steps involved in offloading process, and techniques used in offloading which are discussed below:

3.4.1 Offloading Strategies

In the process of computation offloading, reallocation of massively high computational modules of an application to the remote server or to the other device minimizes the delay in the process of execution and increases the availability of remote resources to all. There are several options for offloading process where IoT devices must contemplate the concerns from various features like when to offload the task, what to offload the task, where to offload the task, and how to offload the task from the end-user's device to the remote server to decrease the latency, energy utilization, time consumption, and bandwidth.

3.4.2 Steps Involved in Offloading Process

The computation offloading process plays an important role in the integration of IoT devices with cloud-edge applications. The following are the three steps that are involved in the offloading process. Figure 2 depicts the flowchart of the offloading process.

- The application partitioning approach plays a major role in computation offloading. In this step, the application is divided into two components, i.e., off loadable and non-off loadable to select which component is to retain with the IoT device and which component is to be offloaded to the remote server to preserve energy and fulfill the task delay. Here, automatic partitioning is the better option when compared with manual partitioning program design because it is very difficult to generalize all the applications. Different partitioning levels affect the number of complications that include compatibility, transfer state, object individuality, performance, and task uploading [56]. Partitioning is of two types: statistically partitioning in which the demanding portions of computation load are divided despite its running state and change in the environmental conditions and dynamic partitioning, the task that is to be offloaded is decided during the runtime process, and it may change according to its current state and environment.
- The preparation step follows all the necessary sets of tasks that an off loadable component is required in IoT applications. Set of tasks like server selection, transfer and installation of code, and execution of tasks of IoT applications.

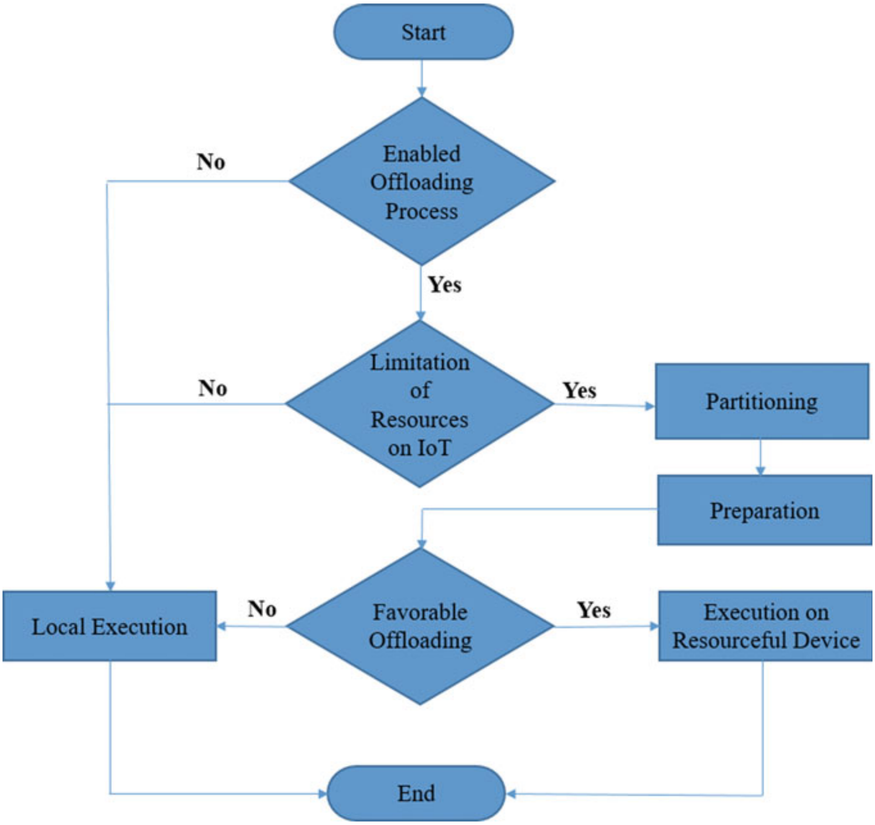


Fig. 2 Offloading process flowchart

- Offloading decision is known as the decision point where it is decided what computation components are to be uploaded to the server for offloading process depending upon the network and running state. The quality of service (QOS) of the process is affected by the precision of its decision-making algorithm. Offloading decision outcomes are local execution where the whole computation is preserved on the local device. The process of offloading data to the remote server and to the other devices have not been accomplished. Partial offloading where the computation has been partitioned into off loadable and non-off loadable components were also used. One component is processed locally at the IoT device, whereas the other component is offloaded to the cloud or any other gateway and full offloading where the whole computation is offloaded and processed by the remote server.

3.4.3 Offloading Techniques

Offloading techniques are categorized as follows:

- **Data Offloading**: This offloading technique is a frequently used offloading technique in which data is further divided into four classes, i.e., data offloading through opportunistic mobile networks, small cell networks, through Wi-Fi networks, and heterogeneous networks. In most IoT devices, the data that is analyzed is only offloaded to the remote server.
- **Method Code Offloading**: In this offloading technique, only the compiler code must be evaluated to identify the relevant code segments for offloading. This technique minimizes network congestion. MAUI and ThinkAir are prominent examples of code offloading [57].
- **Virtual Machine Offloading**: In this offloading technique, the whole operating system with all its running applications is copied to the remote server. It confirms that the running environment is exactly similar to the IoT devices. CloneCloud is commonly used to create a clone of the operating system and its applications to the cloud server, and the offloading is executed using process migration [57].

3.5 Internet of Vehicles (IoV)

Internet of Vehicles is one of the significant aspects of the Internet of Things that creates a worldwide structure for associating the number of smart vehicles and allows these vehicles to interact with one other. IoV is defined as the load on the vehicle electronic sensing devices that is calculated by using the wireless technologies like information network platform that is used for the abstraction and active utilization of the static as well as dynamic information of the vehicle, information technology, and the various functional specifications related to the vehicles state and service. Smart vehicles show features like social driving, safety driving, self-driving, mobile applications, and electric vehicles [58]. It is the developed cyber-physical structure that is formed by the combination of Vehicle Ad hoc Networks (VANETs), mobile cloud computing, and IoT [59]. In IoV, actuators and sensors that are associated with vehicles can interact with Road Side Units (RSUs) [60] and also with additional road users like cyclists and foot-travelers that share and collect information from roads and their surroundings and then make a smart vehicle network that improves the well-being of the traffic [61]. This smart network consists speed of the vehicle, the position of the vehicle route, and other information about the vehicle. The interaction among vehicles has been done through the Internet where all the information is transferred to the central processing unit by using the various components like sensors, infrastructure, Road Side Units (RSU), Global Positioning System (GPS), Radio Frequency Identification (RIFD), image processing through cameras, and other devices, through which the vehicles can complete its environment by collecting the information related to its state. Internet of Vehicles is the global network that empowered vehicles with Wireless Access Technology (WAT) and the Internet through which the various networks infrastructures are used for vehicular communication in IoV. These network infrastructures include vehicle to vehicle (V2V), vehicle to Roadside unit (V2R), vehicle to sensors (V2S), vehicle to infrastructure (V2I), vehicle to personal devices (V2P), and vehicle to everything (V2X). Each network of vehicular communication is enabled with different WAT that includes IEEE WAVE, Wi-Fi, 4G/LTE, MOST/Wi-Fi, and CarPaly/NCF. These architectures include the communication devices associated with vehicles and Road Side Unit.

3.5.1 Architecture of Internet of Vehicles

Internet of Vehicles is considered as the network of connected vehicles that facilitates the communication among vehicles, which includes V2X communication; X may be any vehicle, RSU, infrastructure, personal devices, sensors, and humans. It offers assistance like comfort, convenience, safety, and efficiency to the end-users. IoV system comprises of three parts: terminal of the vehicle, platform that processes the cloud, and the data analysis platform. To connect vehicles with vehicles, vehicles with roads and vehicles with people different technical resources are used. Cloud computing is also used as an information service to complete a huge amount of business applications. The layout of the layered architecture for the global network comprises other different networks for the recognition and combination of components of various networks having the same functions as an individual layer. To do this task, many researchers propose layered architectures. Kaiwartya et al. [62] proposed five-layered architecture in their study that includes perception layer, coordination layer, artificial intelligence layer, application layer, and business layer. Gandotra et al. [63] proposed three-layer architecture in their study where the device-to-device communication is done. Researcher [64] also proposed seven-layer architecture that offers an association among different components of the network and transmission of data in IoV. Authors [65, 66] proposed three-layer architecture that includes the perception, network, and application layer. Long et al. [67] proposed three-layered architecture in their study that consists of the device layer which is the combination of sensing devices and mobile vehicles, mobile edge computing layer (MEC), and cloud center layer. In this architecture, devices transfer the tasks related to the moving vehicles to the MEC servers. The equivalent study of these IoV architectures is subject to the number of layers. But the core architecture of IoV involves the following three layers, namely perception, network, and application layer as mentioned by [65, 66] in their study. The responsibilities and functionalities of each layer are described as follows:

In the architecture of IoV, perception layer is the first layer. It is also recognized as the lowest layer in the architecture. It is associated with the various types of actuators and sensors that are linked with smartphones, RSUs, vehicles, and other types of equipment in the network. To collect the information related to the vehicle like speed, direction, position, acceleration, and condition of the engine related to environments like weather conditions and on-road vehicle density and other information related to devices is the main responsibility of this layer. For the safe and secure data transmission and transformation to the next level perception layer was used and it also have Radio Frequency Identification (RIFD) and perceptions related to satellite positioning. In the architecture of the IoV, network layer is the second layer that is also recognized as the communication layer or coordination layer. It firstly communicates all the data that is collected through sensors in the first layer and transfers the information for processing. At this level, coordination among networks like WLAN, Wi-Fi, GSM, G5, WAVE, 4G/LTE, Bluetooth, and satellite networks is to be represented. To process the various structures of data that are received from networks and to assemble them to form a combined structure that is handled and recognized at every network is the main concern of this layer. In the architecture of the IoV, application layer is the third layer. It is also known as the core layer of the architecture. It is characterized by smart applications, traffic security, efficacy, and infotainment that is based on multimedia. It provides smart services to the users by creating the business cloud by using cloud computing and distributed service centers. The major role of the application layer is to integrate and process the information collected through sensors in the perception layer and then provide information services and applications like the condition of traffic, weather, navigation, etc. to the users.

4 Conclusion

In this paper, an overview of various computation offloading schemes in cloud-edge computing for the Internet of Vehicles is highlighted. To deal with the large number of IoT devices, computation offloading is used, and it offloads the sensitive tasks computationally to remote clouds and improves the resource utilization also decreasing the energy consumption. The aim of the study is to review the various computational offloading schemes that were designed for Internet of Vehicles by different researchers as the part of their study. The review includes articles published in various journals and conference proceedings from 2017 to 2021. Further, it is noted that various offloading schemes are discussed according to the technique used by highlighting the performance metrics of computation offloading in their research work. It has been perceived that cloud computing and edge computing were not considered as the better approach for computation offloading in IoV because that causes various issues like latency, low bandwidth, high energy utilization, load balancing, etc. Hence, due to this reason a hybrid computing technology, cloud-edge computing is considered as the part of the study. This paper also provides an overview of the computing technologies like cloud computing, edge computing, and the hybridization of these technologies. Internet of Vehicles with its proposed architectures by various researchers is also part of the study.

In the future, a computation offloading scheme in cloud-edge computing is to be designed for load balancing, enhancing resource utilization, and minimizing energy consumption. There is an unlimited perspective in the field of the Internet of Vehicles (IoV) because it the fastest growing technology in the family of Internet of Things. Simulation techniques would be used to measure and compare the various computation offloading metrics.

References

1. Kumari A, Tanwar S, Tyagi S, Kumar N, Maasberg M, Choo KKR (2018) Multimedia big data computing and internet of things applications: a taxonomy and process model. J Netw Comput Appl 124:169–195
2. W. Economic Forum (2016) The number of cars worldwide is set to double by 2040. Accessed on June 2021
3. Salahuddin MA, Al-Fuqaha A, Guizani M (2015) Software-defined networking for RSU clouds in support of the internet of vehicles. IEEE Internet Things J 2(2):133
4. Raza S, Wang S, Ahmed M, Anwar MR (2019) A survey on vehicular edge computing: architecture, applications, technical issues, and future directions. Wirel Commun Mob Comput 2:1–19
5. Si-Feng Z, Jiang-Hao C, En-Lin S, Qing-Hua Z (2021) Multi-objective optimized immune algorithm for computing offloading problem in edge computing scenes of internet of vehicles. In: Wireless personal communications. Springer, Berlin
6. He X, Ren Z, Shi C, Fang J (2016) A novel load balancing strategy of software-defined cloud/fog networking in the internet of vehicles. China Commun 2:140–149
7. Wu K, Lu P, Zhu Z, Member S, Model AS (2016) Distributed online scheduling and routing of multicast-oriented tasks for profit-driven cloud computing. IEEE Commun Lett 20(4):684–687
8. Tang W, Zhao X, Rafique W, Qi L, Dou W, Ni Q (2019) An offloading method using decentralized P2P-enabled mobile edge servers in edge computing. J Syst Architect 94:1–13
9. Li Z, Zhou X, Liu Y, Fan C, Wang W (2021) A computation offloading model over collaborative cloud-edge networks with optimal transport theory. In: Published in the proceedings of IEEE 19th International conference on trust, security and privacy in computing and communications (TrustCom)
10. Hu C, Li H, Huo Y, Xiang T, Liao X (2017) Secure and efficient data communication protocol for wireless body area networks. IEEE Trans Multi-Scale Comput Syst 2(2):94
11. Zhang H, Chen S, Li X, Ji H, Du X (2015) Interference management for heterogeneous network with spectral efficiency improvement. IEEE Wirel Commun Mag 22(2):101–107
12. Zhang R, Xie P, Wang C, Liu G, Wan S (2019) Classifying transportation mode and speed from trajectory data via deep multi-scale learning. Comput Netw 162:106861
13. Gao Z, Xuan HZ, Zhang H, Wan S, Choo KKR (2019) Adaptive fusion and category-level dictionary learning model for multi-view human action recognition. IEEE Internet Things J 6(6):9280–9283
14. Wang S, Fan C, Hsu CH, Sun Q, Yang F (2016) A vertical handoff method via self-selection decision tree for internet of vehicles. IEEE Syst J 10(3):1183
15. Cheang CF, Wang Y, Cai Z, Xu G (2018) Multi-VMs intrusion detection for cloud security using Dempster-–Shafer theory. Tech Sci Press 57(2):297–306
16. Xing K, Hu C, Yu J, Cheng X, Zhang F (2017) Mutual privacy preserving k-means clustering in social participatory sensing. IEEE Trans Industr Inf 13(4):2066–2076
17. You C, Huang K, Chae H, Kim BH (2017) Energy-efficient resource allocation for mobile-edge computation offloading. IEEE Trans Wireless Commun 16(3):1397–1411
18. Zheng W, Wang Y, Xia Y, Wu Q, Wu L, Guo K (2017) On dynamic performance estimation of fault-prone infrastructure-as-a-service clouds. Int J Distrib Sensor Networks 13(7)
19. Wang X, Yang LT, Kuang L, Liu X, Zhang Q, Deen MJ (2019) A tensor-based big-data-driven routing recommendation approach for heterogeneous networks. IEEE Network 33:64–69
20. Gopinath V, Bhuvaneswaran R (2018) Design of ECC based secured cloud storage mechanism for transaction rich applications. 99:1–1
21. Zhang K, Mao Y, Leng S, Zhao Q, Li L, Peng X, Pan L, Maharjan S, Zhang Y (2016) Energy efficient offloading for mobile edge computing in 5G heterogeneous networks. IEEE Access 4:5896–5907
22. Xu X, Gu R, Dai F, Qi L, Wan S (2019) Multi-objective computation offloading for internet of vehicles in cloud-edge computing. Wireless Netw 26(3):1611–1629

23. Xu X, Fu S, Yuan Y, Luo Y, Qi L, Lin W, Dou W (2019) Multi-objective computation offloading for workflow management in cloudlet-based mobile cloud using NSGA-II. Comput Intel 35(3):476–495
24. Dai Y, Xu D, Maharjan S, Zhang Y (2018) Joint offloading and resource allocation in vehicular edge computing and networks. In: Proceedings of 2018 IEEE global communications conference (GLOBECOM), pp 1–7
25. Wan S, Xiang L, Yuan X, Wenmin L, Xiaolong X (2019) Efficient computation offloading for internet of vehicles in edge computing-assisted 5G networks. J Supercomput 76:2518–2547
26. Xu X, Yuan X, Lianyong Q, Yuan Y, Xuyun Z, Tariq U, Wan S (2019) An edge computing-enabled computation offloading method with privacy preservation for internet of connected vehicles. Futur Gener Comput Syst 96:89–100
27. Zhang K, Mao Y, Leng S, He Y, Zhang Y (2017) Predictive offloading in cloud-driven vehicles: using mobile-edge computing for a promising network paradigm. IEEE Veh Technol Mag 12(2):36–44
28. Xu X, Xue Y, Li X, Qi L, Wan S (2019) A computation offloading method for edge computing with vehicle-to-everything. IEEE Access 7:131068–131077
29. Zhao J, Li Q, Gong Y, Zhang K (2019) Computation offloading and resource allocation for cloud assisted mobile edge computing in vehicular networks. IEEE Trans Veh Technol 68(8):7944–7956
30. Liwang M, Dai S, Gao Z, Du X, Guizani M, Dai H (2019) A computation offloading incentive mechanism with delay and cost constraints under 5G satellite-ground IoV architecture. IEEE Wirel Commun 26(4):124–132
31. Li B, Peng Z, Hou P, He M, Anisetti M, Jeon G (2019) Reliability and Capability based computation offloading strategy for vehicular Ad Hoc clouds. J Cloud Comput 8(1)
32. Lin L, Xiaofei L, Hai J, Peng L (2019) Computation offloading toward edge computing. In: Proceedings of the IEEE, pp 1–24
33. Hou X, Zhiyuan R, Jingjing W, Wenchi C, Yong R, Cheng CK, Hailin Z (2020) Reliable computation offloading for edge computing-enabled software-defined IoV. IEEE Internet Things J 4662(c): 1–1
34. Xu X, Zhang X, Liu X, Jiang J, Qi L, Bhuiyan MZA (2020) Adaptive computation offloading with edge for 5G-envisioned internet of connected vehicles. IEEE Trans Intell Transp Syst, pp 1–10
35. Sun J, Gu Q, Zheng T, Dong P, Valera A, Qin Y (2020) Joint optimization of computation offloading and task scheduling in vehicular edge computing networks. IEEE Access 8:10466–10477
36. Wang G, Xu F (2020) Regional intelligent resource allocation in mobile edge computing based vehicular network. IEEE Access 8:7173–7182
37. Chen C, Chen L, Liu L, He S, Yuan X, Lan D, Chen Z (2020) Delay-optimized V2V-based computation offloading in urban vehicular edge computing and networks. IEEE Access 8:18863–18873
38. Wang Z, Zheng S, Ge Q, Li K (2020) Online offloading scheduling and resource allocation algorithms for vehicular edge computing system. IEEE Access 8:52428–52442
39. Ning Z, Huang J, Wang X, Rodrigues JPC, Guo L (2019) Mobile edge computing-enabled internet of vehicles: toward energy-efficient scheduling. IEEE Network 33(5):198–205
40. Garg S, Singh A, Kaur K, Aujla GS, Batra S, Kumar N, Obaidat MS (2019) Edge computing-based security framework for big data analytics in VANETs. IEEE Network 33(2):72–81
41. Arunarani AR, Manjula D, Sugumaran V (2019) Task scheduling techniques in cloud computing: a literature survey. Futur Gener Comput Syst 91:407–415
42. Keshanchi B, Souri A, Jafari N (2017) An improved genetic algorithm for task scheduling in the cloud environments using the priority queues: formal verification, simulation, and statistical testing. J Syst Softw 124:1–21
43. Yousefpour A et al (2019) All one needs to know about fog computing and related edge computing paradigms: a complete survey. J Syst Architect 98:289–330

44. Li W, Chen Z, Gao X, Liu W, Wang J (2019) Multi-model framework for indoor localization under mobile edge computing environment. IEEE Internet Things J 6(3):4844–4853
45. Wang M, Wu J, Li G, Li J, Li Q, Wang S (2017) Toward mobility support for information-centric IoV in the smart city using fog computing. In: 5th International proceedings of the smart energy grid engineering conference (SEGE), pp 357–361
46. Shi W, Cao J, Zhang Q, Li Y, Xu L (2016) Edge computing: vision and challenges. IEEE Internet Things Journal 3(5):637–646
47. Hadzic I, Abe Y, Woithe H (2017) Edge computing in the ePC. In: Proceedings of the Second ACM/IEEE symposium on edge computing - SEC '17
48. Baktir AC, Ozgovde A, Ersoy C (2017) How can edge computing benefit from software-defined networking: a survey, use cases, and future directions. IEEE Commun Surv Tutorials 19(4):2359–2391
49. Jiang L, Chang X, Yang R, Jelena M, Vojislav B (2020) Model-based comparison of cloud-edge computing resource allocation policies. Oxford Univ Press, Standard J Pub 63(10):1564–1583
50. Wang X, Yang LT, Xia X, Jin J, Deen MJ (2017) A cloud-edge computing framework for cyber-physical–social services. IEEE Commun Mag 55(11):80–85
51. Hu P, Dhelim S, Ning H, Qiu T (2017) Survey on fog computing: architecture, key technologies, applications and open issues. J Netw Comput Appl 98:27–42
52. Shaukat U, Ahmed E, Anwar Z, Xia F (2016) Cloudlet deployment in local wireless networks: motivation, architectures, applications and open challenges. J Netw Comput Appl 62:18–40
53. Tseng FH, Cho HH, Chang KD, Li JC, Shih TK (2018) Application-oriented offloading in heterogeneous networks for mobile cloud computing. Enterprise Inf Syst 12(4):398–413
54. Peng H, Wen WS, Tseng ML, Li LL (2019) Joint optimization method for task scheduling time and energy consumption in mobile cloud computing environment. Appl Soft Comput 80:534–545
55. Xu X, He C, Xu Z, Qi L, Wan S, Bhuiyan MZA (2019) Joint optimization of offloading utility and privacy for edge computing enabled IoT. IEEE Internet Things J 7(4):2622–2629
56. Akherfi K, Gerndt M, Harroud H (2018) Mobile cloud computing for computation offloading: issues and challenges. Appl Comput Inf 14(1):1–16
57. Meurisch C, Gedeon J, Nguyen TAB, Kaup F, Mühlhäuser M (2017) Decision support for computational offloading by probing unknown services. In: 2017 26th International conference on computer communications and networks, ICCCN 2017
58. Maglaras LA, Al-Bayatti AH, He Y, Wagner I, Janicke H (2016) Social internet of vehicles for smart cities. J Sens Actuator Networks 5(3)
59. Vashi S, Ram J, Modi J, Verma S, Prakash DC (2017) Internet of things (IoT): a vision, architectural elements, and security issues. In: Conference: 2017 International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), vol 1, pp 492–496
60. Bonomi F (2013) The smart and connected vehicle and the internet of things, Synchronization Telecommun Syst, pp 1–53
61. Singh D, Singh M (2015) Internet of vehicles for smart and safe driving. In: Proceedings of international conference on connected vehicles and expo (ICCVE). IEEE, New York, pp 328–329
62. Kaiwartya O, Abdullah AH, Cao Y, Altameem A, Parsad M, Lin C, Liu X (2016) Internet of vehicles: motivation, layered architecture, network model, challenges, and future aspects. IEEE Access 4:5356–5373
63. Gandotra P, Kumar Jha R, Jain S (2017) A survey on device-to-device (D2D) communication: architecture and security issues. J Netw Comput Appl 78:9–29
64. Castillo JC, Zeadally S, Guerrero Ibanez JA (2017) A seven-layered model architecture for internet of vehicles. J Inf Telecommun 1(1):4–22

65. Sadiku MNO, Tembely M, Musa SM (2018) Internet of vehicles: an introduction. Int J Adv Res Comput Sci Software Eng 8(1):11
66. Ying P (2019) Research on the development of internet of vehicles technology. Internet Things Cloud Comput 7(1):12
67. Long J, Luo Y, Zhu X, Luo E, Huang M (2020) Computation Offloading through mobile vehicles in IoT-edge-cloud network. J Wireless Com Network 2020:244

A Review on Machine Learning-Based Patient Scanning, Visualization, and Monitoring

Ahmed Al Ahdal and Priyanka Chawla

Abstract One of the most important topics for society is human health care; to find the appropriate diagnose or correct diseases, detection is the primary key to get appropriate care; traditional technique is facing many challenges from delay or unnecessary treatment to incorrect diagnoses which lead to a diagnostic error that can effect on the treatment progress, increasing the bill, and give more time to the disease to spread or affect and harm the patient body. Those such errors could be avoided and minimized by using machine learning algorithms. In recent years, many significant efforts have indeed been developed to increase computer-aided diagnosis detection applications, which is a rapidly increasing area of research, and machine learning algorithms are particularly significant in CAD, which is used to detect patterns from medical data sources and making nontrivial predictions could assist the doctor and clinical in making decisions on time. This paper will discuss different ML algorithms that are used in diagnosing different diseases. Therefore, in this paper two major diseases have been chosen like cancer and heart disease, and the use of several ML algorithms applied their performance and accuracy.

Keywords Machine learning technique · Diagnostic system · Human diseases · Machine learning

1 Introduction

In the past, humans discovered many diseases, either simple or infectious, and treated them in simple traditional ways such as using plants, leaves, and roots or by cauterization. As time passed, some treatments appeared using new methods such as ointments, and with the emergence of chemistry and several scientists specialized in this study, they discovered many treatments for incurable or infectious diseases, and they were able to treat those disease such as smallpox disease. As the world and planet have changed drastically over the past few decades due to factors such as the high

A. Al Ahdal · P. Chawla (✉)
Lovely Professional University, Jalandhar 144411, India
e-mail: priyankachawla.cse@gmail.com

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_40

consumption of natural resources, rises in gases, emissions from factories and automobiles, wars and the remnants of those conflicts in which various types of chemical or biological weapons were used, the amount of toxins in food, the expansion of the ozone hole in the atmosphere layers, and the ability of harmful radiation from space to penetrate the atmosphere layers and reach the Earth. This has a direct impact on human health and living environment. All of these and other factors contributed primarily to the change in the shape of human genetic genes, which resulted in the emergence of many diseases that did not exist previously and are very difficult to treat using traditional methods or tools and technology available several years ago. There are still numerous illnesses that are difficult to diagnose or treat, such as cancer, which causes over 200 different types of fatalities in individuals of all ages each year and over 10 million deaths annually, or heart disease, which causes 17.9 million deaths annually [1].

1.1 Motivation

For the reasons stated in the introduction part, traditional methods and techniques are incapable of detecting or treating current diseases; as a result, it is critical to transition to AI and ML technology because of its ability to deal with massive amount of data and its ability to observe beyond the capacity of human brain and predict the result with accurate score, and AI is a tool that mimics actions of human brain performance on a daily basis but much more intelligently that has the huge ability for storing information, classifying, observing, and predicting with high accuracy like problem solving, pattern spotting, and knowledge acquisitions. Motivated by this potential, this study emphasizes that AI and ML technology can offer significant tools and make a huge difference in the treatment of human disease. Consequently, unfortunately only a few researchers are working on it.

1.2 Contribution

- If a comprehensive analysis was carried out to investigate in-depth various existing machine learning algorithm, techniques and methods predicated on human diseases scanning, visualizing, and monitoring of patients
- Various machine learning approaches and methodologies have been compared and classified based on their characteristics, performance, and accuracy.
- Existing studies have been reviewed as well as addressed, and future research directions in the areas of AI and ML in health care have been highlighted.

1.3 Organization

The work presented in this paper is organized as follows: Sect. 2 discusses background theory which presents brief review about machine learning algorithm; Table 1 shows comparison of all machine learning types with brief definitions and example, the ML model, and working; Sect. 3 discusses machine learning in health care; Sect. 4 highlights related research on cancer and heart disease, along with a comparison table for the various machine learning approaches employed; Sect. 5 presents research methods that discussed research question, source information, and search criteria; and finally conclusion is given in Sect. 6.

2 Background Theory

Machine learning algorithm is an area in AI that makes it possible for computers to learn from their previous experiences. In order to identify previous undiscovered patterns and relationships between different features in the dataset. ML focuses on using data and algorithm to mimic the way of human learning. Mishra and Shukla [2] Machine learning is considered as the backbone of big data and data science, there are many applications and technologies that have been successfully applied in different field from pattern recognition, competitional biology, and computer vision to medical field, and there are many machine learning application in health care from disease detection and diagnosis such as QuantX which is a machine learning algorithm and artificial intelligence tool to address the patients' needs and help patient administrator by providing information that helps in making diagnosis with accurate result faster than the traditional technique. References [3, 4] suggest necessary steps to prevent diseases. To solve problems with medical imaging, customised medicine, smart health records, and disease prediction, machine learning is utilised in drug production and discovery. It uses data science to observe and build biological models. Massive amounts of data are applied to by artificial neural networks (ANN), which then use the data's creation to create predictions. Here are a few ML applications in the medical sector [5]. Machine learning has been widely divided into various form as shown in Fig. 1.

Machine learning types such as supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, and deep learning has been covered in detail in Table 1, which includes definitions, types, working examples, and sample graphs.

As shown in Fig. 2, machine learning works by feeding the model training data made up of input features (predictor) and outputs (target). The model then learns how to map input to output and generalises to work even on unseen data.

As in Fig. 2, the following steps are included in the machine learning process:

Data Collection—Gathering information from different perspectives is the very first stage in the machine learning model [6].

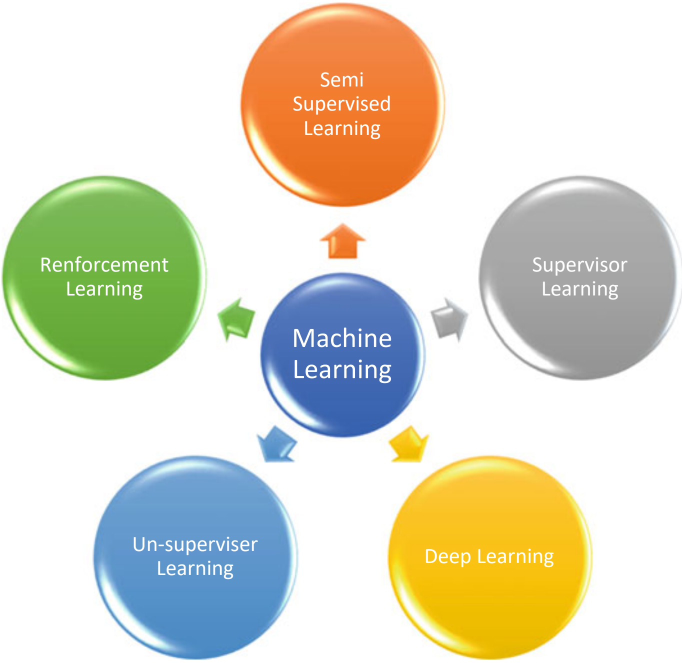


Fig. 1 Machine learning types

Data Preparation—After the data has been gathered, it is recognized, sorted, and classed before it can be analyzed. The data preprocessing strategies used are determined by the type of actions required by the machine learning model.

Training—This step includes allowing the algorithm to self-learn from analyzed data. Training techniques are applied based on the application's different parameters and desired outcomes [7].

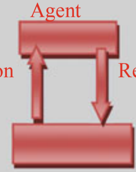
Evaluation—The machine learning program is verified in this stage to measure its achievement, discover errors, and recommend improvements [8].

Making adjustments—Developing machine learning techniques is a never-ending task. The algorithms and the machine learning application model must be quite well as parameters were tested and analytical techniques grow [9].

3 Machine Learning in Health Care

It is quite tough to build up in an exact manner a big data report and clinical diagnostic of the patient's cure and treatment; otherwise, it will be impacted due to inadequate storage or management. This amount of data needs special means or tools to extract and process efficiently, by using one of the machine learning applications such as

Table 1 Machine learning types

Supervised learning	Unsupervised learning	Semi supervide learning	Renforcement learning	Deep learning
1.Labelled data is provided 2.Data with clearly output is given 3.Direct feedback is given 4. Algorithm trains to improve outcome over time Types: 1.Classification To predict district value (TRUE ,FALSE)(1,0) 2.Regerssion To predict continouse value (price) Example: A- NN (Nearst Neighbor) B. NB (Naive Bayes) C. DT (Decision Tree) D. SVM (Support Vector Machine) E. LR (Linear Regerssion) F. Nns (Neaural Network)  	1-unlabelled data is provided 2.Not train data and trarget are not provided 3.No prior knowledge 4-using DL to identify pattern in data and find the similirites between input data and base on those similirities classify the data Types: 1.Clastering Makes cluster or grouping based on similarity 2.Association Identify sequences 3.Dimensionalityreduction Widenvironmenter dependencies Example: A. K means B. PCA D. SVD  	1.Is class of supervised 2.Used unlabeled data for training 3.Combination between minimum amount of labeled data with huge amount of unlabeled data 4.Combinietion of supervised and unsupervised learning Types: 1.Classification 2.Clastering Example: A.Self training B.Mixture models C.Seme suervised SVM  	1.Input or output set are not providedonly agent intracting with environment 2.It has to explore and test all verious possibilities until it finds the right answer 3.Reward based learnng. 4.Maximize the reward. Types: 1.Q learning 2.TD learning 3.Gradient descen  	1.Is subset of machine learning inspired by human brain 2. Based on artificial neural networks 3.Have the ability to train it self and perform high accuracy 4.Use several layer of neural network, larg amout of data and more time of training Types: 1.Generative models 2.Hybrid architecture 3.Discriminative model Example: 1.Deep neural networks(dnn) 2.Convolutional NN (CNN) 3.Deep Q network 4.Recurrent NN (RNN)

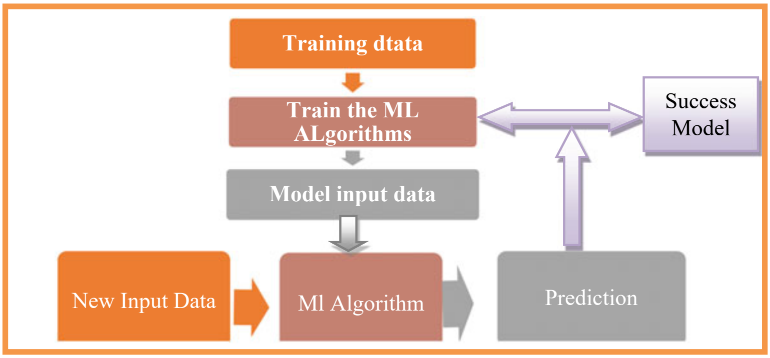


Fig. 2 Machine learning model working

classifier which can divide the data according to their attributes, and this can be used in medical data analysis or disease detection [10].

ML was initially created to analyse medical data sets; over the past few years, ML technologies have made significant progress in the field of disease diagnosis; several reports and records from various modern hospitals demonstrate the effectiveness of ML technologies' output; ML has long way from the past which offered speech recognitions, self-driving cars, and efficient web search to today; ML is present everywhere, and one can use it many times a day; in medical field, it is used in various disciplines such as drug discovery, helps in complex surgeries to assisting surgeons, and provides electronic health record (HER) which is alternate opinion for prediction [11].

4 Related Work

To gain a better understanding of what has been accomplished as well as the difficulties that ML and AI have encountered in order to detect and diagnose diseases, we compared our findings to all prior research, compared them to each other, and then we analysed the findings.

4.1 Cancer Disease

According to doctors and clinicians, there is no cure for cancer yet but there is a treatment that may be able to cure some people for some cancer, some types of cancer grow slowly, others spread very quickly, every kind of cancer is different, and there is no one size or one symptom. Early detection of cancer increases the patient's chances of survival significantly, according to all researchers and oncologists, and there are no better treatment alternatives for either preventing the metastasis or completely removing the tumor [12]. New technologies, particularly artificial intelligence, machine learning, and deep learning, should be employed as a diagnostic tool. As a diagnostic tool, very complex software's diagnostic abilities have been tested and compared to traditional diagnostic tools and expertise and shown to be extremely useful in disease diagnosis and prognosis, and the below is the latest paper in field of machine learning in cancer [13].

The proposed methodology eliminates the likelihood of errors in the manual approach by using DL approaches, notably CNNs, according to Kumar et al. [14] while detecting Blood cancer with a score of 97.2% accuracy. The algorithm, which had been engaged on images of cellular, firstly preprocesses the images and takes the best attributes.

Naqi et al. [15], DL and score accuracy for lung cancer was 96.9%. This study seeks to offer a thorough detection mechanism and categorization to aid clinicians in their diagnosis because the system has a difficulty producing false-positive results.

Senturk et al. [16], in breast cancer using SVM and score 96.40% accuracy, find the most efficient techniques on early detection of breast cancer. A comprehensive review of breast cancer diagnosis in patients is provided [17]. Table 2 shows different research papers in last 10 years which is used in various cancer detection or diagnoses using different machine learning algorithms.

Figure 3 shows each paper related to different type of cancer up to 2020, and different ML techniques are used with the author name, year of publication, and how much accuracy is given as we see in this Fig. 3. There is a bright future for detecting

Table 2 Machine learning technique in cancer diseases

Author	Disease	Methods	Year	Accuracy
Horie et al.	Esophageal cancer	CNN	2019	98%
Ganggayah et al.	Breast cancer	RF	2019	82.7%
Kumar et al.	Blood cancer	CNN	2020	97.2%
Naqi et al.	Lung cancer	DL	2020	96.9%
Cinarer et al.	Brain tumor	KNN, RF, SVM and LDA	2019	90% SVM

Table 3 Machine learning technique in heart diseases

Author	Disease	Methods	Year	Accuracy (%)
Shan Xu et al.	Heart disease	SVM	2017	98.9
Kamran Farooq et al.	Heart disease	DT	2014	78.4604
Otoom et al.	Heat disease	SVM	2015	88.3
Syed Muhammad et al.	Heat disease	SVM	2017	91.30
MeghaShahi et al.	Heat disease	SVM	2017	85

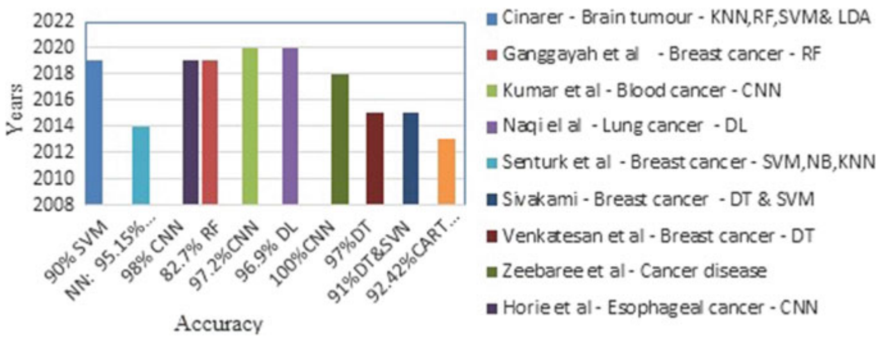


Fig. 3 Summary of ML technique for cancer disease detection

cancer which is the main factor for curing; if the cancer gets detected in early stage, there is a big chance that it will be cured with proper medications.

4.2 Heart Disease

Heart disease is one of the leading causes of the death in the world, roughly one person lost his or her life per minute due to heart diseases. The main challenge is predicting the disease in its early stages, which is why the majority of studies and research on health care focus on machine learning, which has the ability to detect with accurate results. The table below demonstrates how machine learning algorithms have been used most frequently for heart disease with their accuracy. The following is latest work related to heart disease using machine learning algorithm.

Senthil Kumar et al. [17] improved prediction of cardiovascular disease using composite machine learning techniques that includes a methodology which seeks to find significant implication through applying machine learning, leading to enhanced accuracy within prediction of cardiovascular illness.

Vembandasamy et al. [18] work was performed by using Naïve Bayes algorithm which is a powerful independence assumption, the data was obtained from diabetic research institute, and it consists of 500 patients' record. Naïve Bayes algorithm offers 86.919% of accuracy. Table 3 shows different ML technique in heart disease.

As we see in Table 3, the most of machine learning algorithm has been used for heart disease with their author, year, and accuracy. There are many other studies, but our aim was for latest paper from last 10 years [19]. The main idea after reviewing the above tables and the purpose of all articles was to compare the accuracy and F-measure scores of classification methods such as Decision Tree, Random Forest, Logistic Regression, and Naive Bayes, and to effectively predict if the patient suffers from heart disease or not having any heart-related illness.

Table 4 List of keywords

Keywords	Synonyms
SVM	Support vector machine
NB	Naïve Bayes
DL	Deep learning

5 Research Methods

5.1 Research Questions

This objective of the paper is to highlight the most cutting-edge developments in machine learning algorithms that answer the following research questions: 1—What is machine learning and its facets? What is the need of machine learning in medical field?

2— What are the characteristics of machine learning algorithms? 3—What are the different approach and techniques used in machine learning algorithms in health care? 4—What are the limitations and benefits of machine learning algorithms in medical field?

5.2 Source of Information

The collection and selection of paper processed and done in the way which meets a predefined criterion; research paper was published by IEEE, Elsevier, Nature, and ACM. The databases covered are IEEE–Springer–Nature (https://www.nature.com/)—MEDLINE (PubMed)—ScienceDirect4.3.

5.3 Search Criteria

As search terms, a collection of key words and phrases has been defined based on the research questions. Because machine learning research is still in its early stages, no particular year scope was included in the search. Table 4 also includes a list of the keywords that were used.

6 Conclusion

Traditionally, cancer detection and treatment have been determined based on the expertise of a pathologist; those clinicians are already working in the health sector for more than fifteen years and have examined variety of patients having comparable symptoms, yet their precision is still not 100% with the growth of artificial intelligence and machine learning. The detection of certain tumours has been facilitated by artificial intelligence and machine learning technologies. Various reports and records from different modern hospitals indicate the efficiency of ML technologies' result. Many applications use artificial intelligence to detect cancer diseases, but there are no conclusive results for eliminating this disease due to its evolution and different types. The same thing for predicting and detecting heart-related disease using ML and AI is generally quite significant based on the above work, and it is possible to conclude that machine learning algorithms have a large potential for predicting and diagnosing cancer and cardiovascular illnesses or any heart-related diseases.

References

1. https://www.who.int/health-topics/cardiovascular-diseases
2. Mishra A, Shukla A (2018, December) From machine learning to deep learning trends and challenges. CSI Communications
3. Allix NM, Epistemology and knowledge management concepts and practices. J Knowl Manage Pract
4. Anil Mathur GP (2007) Socialization influences on preparation for later life. J Market Pract: Appl Market Sci 5:163–176
5. McConnell KJ, Lindner S (2019) Estimating treatment effects with machine learning. Health Serv Res 54(6):1273–1282
6. Maity NG, Das S (2017) Machine learning for improved diagnosis and prognosis in healthcare. In: IEEE aerospace conference proceedings, Big Sky, MT, USA, pp 1–9
7. National Center for Biotechnology Information. http://www.ncbi.nlm.nih.gov
8. Fatima M, Pasha M (2017) Survey of machine learning algorithms for disease diagnostic. J Intell Learn Syst Appl 09(01):1–16. https://doi.org/10.4236/jilsa.2017.91001
9. Bhavsar KA, Abugabah A, Singla J, AlZubi AA, Bashir AK (2021) A comprehensive review on medical diagnosis using machine learning. Comput Mater Contin 67(2):1997–2014. https://doi.org/10.32604/cmc.2021.014943
10. Vohra R, Rani P (2014) Liver patient classification using intelligent techniques. Int J Comput Sci Inf Technol (IJCSIT) 5:5110–5115
11. Rajeswari P, Reena GS (2010) Analysis of liver disorder using data mining algorithm. Global J Comp Sci Technol 10:48–52
12. Tarmizi NDA, Jamaluddin F, Abu Bakar A, Othman ZA, Zainudin S, Hamdan AR (2013) Malaysia dengue outbreak detection using data mining models. J Next Gener Inf Technol (JNIT) 4:96–107
13. Rajkumar SV, Kumar S (2020) Multiple myeloma current treatment algorithms. Blood Cancer J 10(9):94. https://doi.org/10.1038/s41408-020-00359-2
14. Kumar D, Jain N, Khurana A, Mittal S, Satapathy SC, Senkerik R, Hemanth JD (2020) Automatic detection of white blood cancer from bone marrow microscopic images using convolutional neural networks. IEEE Access 8(Mm):142521–142531. https://doi.org/10.1109/ACCESS.2020.3012292
15. Naqi SM, Sharif M, Jaffar A (2020) Lung nodule detection and classification based on geometric fit in parametric form and deep learning. Neural Comput Appl 32(9):4629–4647. https://doi.org/10.1007/s00521-018-3773-x
16. Senturk ZK, Kara R (2014) Breast cancer diagnosis via data mining: performance analysis of seven different algorithms. Comput Sci Eng 4(1):35

17. Mohan S, Thirumalai C, Srivastava G (2019) Effective heart disease prediction using hybrid machine learning techniques. IEEE Access 7. https://doi.org/10.1109/ACCESS.2019.2923707. Bingulac SP (1994) On the compatibility of adaptive controllers. In: Proceedings of the Fourth Annual Allerton Conference on Circuits and Systems Theory, pp 8–16
18. Vembandasamy K, Sasipriya R, Deepa E (2015) Heart diseases detection using Naive Bayes algorithm. IJISET—Int J Innov Sci Eng Technol 2(9):441–444
19. Otoom AF, Abdallah EE, Kilani Y, Kefaye A, Ashour M (2015) Effective diagnosis and monitoring of heart disease. Int J Softw Eng Appl 9:143–456
20. Xu S, Zhu T, Zang Z, Wang D, Hu J, Duan X et al (2017) Cardiovascular risk prediction method based on CFS subset evaluation and random forest classification framework. In: 2017 IEEE 2nd International Conference on Big Data Analysis

Natural Language-Based Naive Bayes Classifier Model for Sentence Classification

Amita Yadav, Sonia Rathee, Shalu, and Sherin Zafar

Abstract The classification of text is one of the basic tasks of natural language processing with wide-ranging applications. This is essentially a process of assigning markers or categories to the text based on its content. The paper aims to use an improved Naive Bayes classifier to identify the fact-worthy sentence. In this paper, authors have implemented an improved Naive Bayes classifier through which we classify the sentences. This proposed method has been tested with the claim buster dataset contains 23,533 sentences where each sentence belongs to either of these three classes, i.e., non-factual statement, unimportant factual statement, and check-worthy factual statement.

Keywords Text classification · Naïve Bayes · Machine learning algorithm · Sentence detection

1 Introduction

Text classification is a technique useful in the detection of spam, data mining, sentiment analysis. Text classification is a method to add tags to the texts with the applicable predefined category in dataset. In this paper, we implemented an improved Naive Bayes classifier through which we classify the sentences. So, we used a text classification mechanism to implement an improved Naive Bayes algorithm. Before applying any machine learning model on text data, feature plays an important role in it. And for the text dataset, the feature is also a text which is present in the dataset. As we know that Naive Bayes classifier assumes that features are independent of each other. And that is one of the worst assumptions or demerits of Naive Bayes for highly correlated features. The proposed work also came up with the solution of zero handle

A. Yadav · S. Rathee · Shalu
Maharaja Surajmal Institute of Technology, New Delhi, India

S. Zafar (✉)
Jamia Hamdard, New Delhi, India
e-mail: sherin.zafar@jamiahamdard.ac.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_41

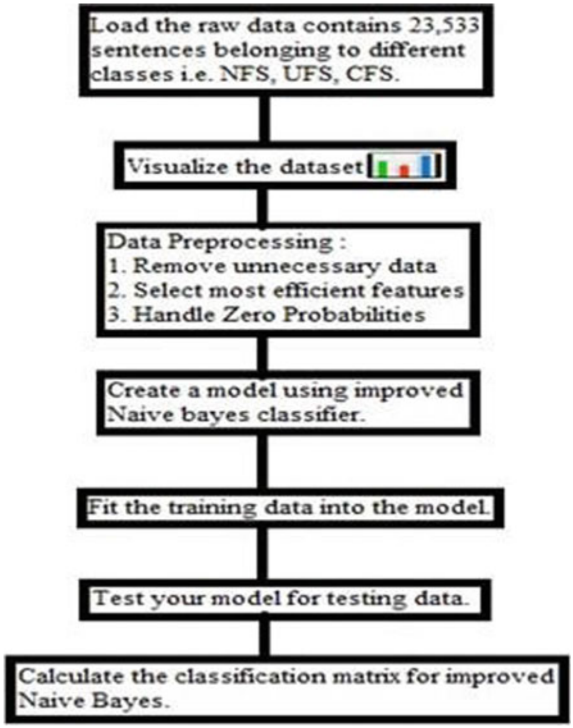


Fig. 1 Flow diagram

probability problems which were generally observed in Naive Bayes classifier when a testing data are different from training data; then, it gives zero probability to testing data and no class is assigned to testing data. So, here, we used Laplace correction to assign the class to testing data. The working process is given in Fig. 1.

For any text classification, following step is utilized as:

- Data Collection: The first step is selecting a dataset and defining the categories in which data will be divided. The approach used a claim buster dataset which is publicly available. This dataset contains the claims from US Presidential Debate. This dataset contains around 25 thousand sentences, and each sentence is assigned one of the three categories, i.e., unimportant factual statement, non-factual statement, and check worthy factual statement.
- Preprocessing: Preprocessing of data is the process of extraction of useful or desired information from raw data in the dataset. For text processing, that includes stemming, text cleaning, stop words removal, and lemmatization. Stop words are the words having grammatical importance but do not add any significant meaning in terms of categorization. (e.g., “that,” “the,” “an,” “was,” “an,” “and”). They are not used in decision-making because they are generally equally distributed in documents of various categories. The method of removing stop words is very

straightforward; the words present in documents are scanned and compared with the list of stop words and then remove those words present in the stop list.

- Feature Selection: The main idea of feature selection is removing redundant data from documents by preserving the information of the document. Feature selection helps in the easy interpretation of models and reduces the time required for training of dataset. The purpose of feature selection is to take those features from the dataset which can define your model perfectly and efficiently.
- Handle Zero Probability: The effectiveness of a Naive Bayes classifier gets decreased if the dataset has a significant variation with the attributed distribution.

One example is the condition where the value of the categorical attribute is not found in the dataset, and the result is the model incompetent to make a forecast because of allocation of 0 probability. For the improvement, the model needs to get reformed and get the recent computation of probability after examining and manipulating these cases. In the research paper, further, Sect. 2 defines literature review. Further, Sect. 3 discusses methodology specifying the process of Naïve Bayes classifiers. Sections 4 and 5 present simulation and result analysis. The research paper lastly discusses the conclusion and future scope under Sect. 5. References and the resources utilized during the research paper construction are enlisted under title references.

2 Literature Review

Naive Bayes algorithm have been presented where zero conditional probability problem arises and ways to solve it are some problems of this algorithm. Class conditional probability estimation is a major problem in Naive Bayes model and the general way for that is kernel density estimation as discussed in [1, 2]. Rish [3] explains Naive Bayes which is simply Bayesian classifier with belief of independency among features is competitive with state-of-the-arts classifiers like C4.5, and its performance is affected by data characteristics which are our broad aim. Ikonomakis and Kotsiantis discussed that even for a specified classified method, it is observed that classification performances of the corpuses of the classifiers are different based on different training text. This implies that in sometimes, classifier performance is pertinent to its training corpus, and classifiers of good performance can be derived from high-quality training corpuses [5]. Stephens et al. describe that for sturdy performance over a wide spectrum of query Naive Bayes approximation is commonly used [4]. Nakov [6] was intended about predicting claims in a political debate that should be a priority for fact-checking. A worth-based ranking list for fact-checking was produced. Approaches explored: KNN, SVMs, random forests, and re-current and multi-layer neural networks. The dataset is extremely small in size. It is confined to the political area alone. Adding annotations from different sources and expanding the corpus could yield better results. Kamran Kowsari, et al. targeted to check the political area and consequently verify factual claims. Accord between various sources was less so diligent task is needed to get the better of prejudice in the categorization of claims, so the corpus requires to incorporate more debates, interviews to be expanded [7]. Li et al. work to recognize the crucial claims by juxtaposing the output with CNN and Politifact. Final affirmation is yet required by humans because wholly preprogrammed ways for fact-checking still be inadequate in respect of quality [8]. Patil et al. discussed that for increasing the precisions of any text-based machine learning algorithms, preprocessing techniques form a huge contribution. TF–IDF and bag-of-words model are commonly used feature extraction technique for down-scaling the irrelevant features, and highlighting the prominent features TF-IDF is by far the perfect choice [9]. Document classification is a notable and well-studied domain of pattern recognition with an assortment of new era applications. In this paper, the achievement of the algorithm is checked using numerous classifiers by putting forward a term frequency with stemmer-based attribute extraction algorithm, and it is seen that the proposed technique outperforms other techniques discussed in [10].

3 Proposed Work

3.1 Data Set Description

The dataset comprises of 23,533 sentences, and every statement is classified into—Check-worthy factual claim, non-factual claim, and unimportant factual claim. We present a dataset of claims from all US presidential debate (1960–2016). The data are developed using the statements provided by presidential members during past presidential debates. The dataset includes three CSV files: grouthtruth.csv: It contains testing sentences whose labels were settled upon by three specialists, crowd-sourced.csv: It comprises sentences that were labeled by top-quality members and allsentences.csv: It contains not only labeled sentences but all presidential debate sentences. The data of both the ground-truth and crowd-sourced files are involved in a similar list of attributes, listed below:—Sentence Id—a unique integral identifier to distinguish sentences in the dataset. Text—a sentence that was delivered by a debate member. Speaker—name of the person who was delivering the sentence provided in the text. Speaker Title—speaker's designation at the time of debate. Speaker Party—the political affiliation of speaker. File Id—debate record identifier. Length—he number of words in the sentences provided for the text. Line Number—an integral identifier to signify the order of the sentences in text column according to the debate transcript. Sentiment—sentiment represented a score, which ranges from -1 to $+1$. -1 represents the most negative sentiment, while $+1$ represents the most positive sentiment. Verdict—labels have been assigned to the classes: For CFS, it is assigned 1; for UFS, it is assigned 0, and for NFS, it is -1.

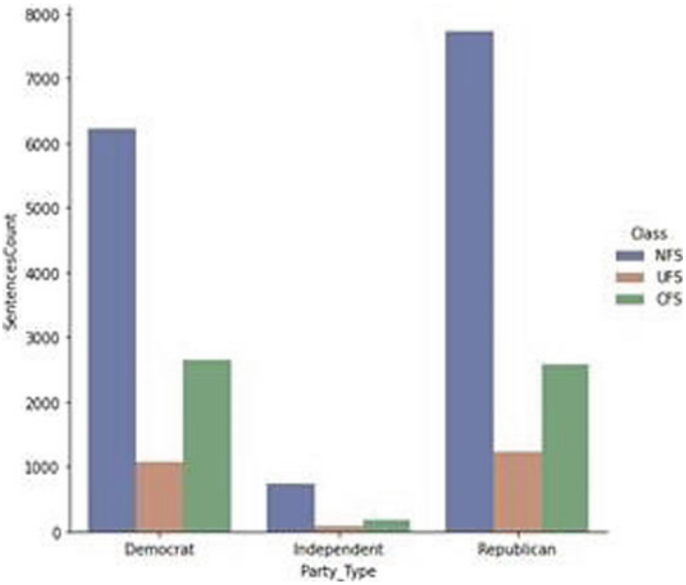


Fig. 2 Claim distribution for party type

3.2 Data Visualization

Given the amount of information gathered by data analysis in today's world, we need to have a way to paint a picture of that data to interpret it. Data visualization gives us a clear picture of what information means by providing visual context with maps or graphs. To understand the behavior (or pattern) of data, we need to first visualizeit. The dataset contains the sentences, spoken during 33 US presidential debates from 1960 to 2016. There was a total of 69 presidential members in the general election debates. These candidates were the part of three different political parties: Republican, Democrat, and Independent Party. The division of these was such that 33 out of 69 was from Republic Party, 32 was from Democrat Party, and rest were from independent candidates. Figure 2 will represent the distribution of claims over the speaker party type.

Figure 3 will be representing the distribution of claims over speaker's designation, while Fig. 4 represents the claim distribution over speakers in the elections.

3.3 Data Cleaning and Feature Selection

You cannot directly apply the raw text to any machine learning model. First, you need to clean up your text, which means dividing it into words and dealing with punctuation and case. We used different techniques to clean the text present in the

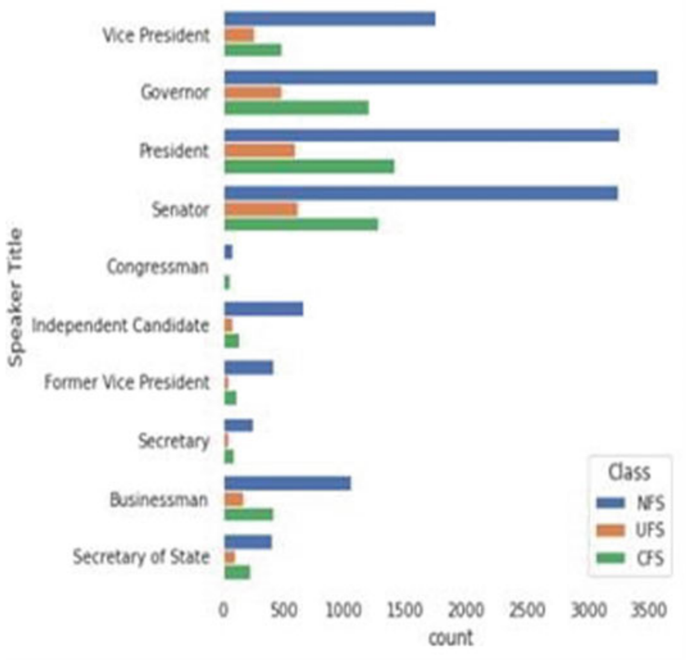


Fig. 3 Claim distribution over speaker title

dataset. We used the count vectorizer tool to remove stop words (i.e., is, am, the) from our document. The method of removing stop word is very straightforward; the words present in documents are scanned and compared with list of a stop words and then remove those words present in stop list. Stemming is a process where various forms of a word mapped to a single word. Any stemming algorithm will convert different forms of tokenized word into a root word.

3.4 Improvement on Naive Bayes Classifier

In Naive Bayes, the probability of each feature is calculated independently from the training dataset. The reason to call it Naive because it is based on the assumption that each feature is independent of each other. We can use a variety of algorithms to explore the combination of probabilities of various features together and evaluate their performance at predicting the output variable. After combining features with each other and removing unnecessary features, we have a new set of efficient features.

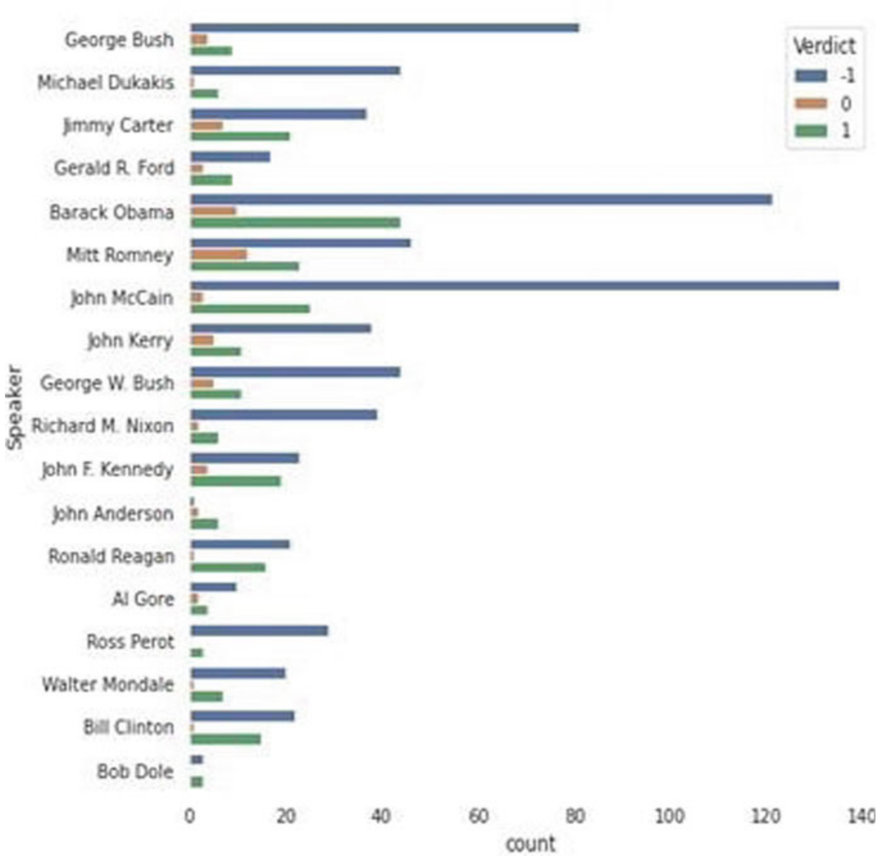


Fig. 4 Claim distribution over speaker

3.5 Naive Bayes Classifier

Naive Bayes is a statistical ranking technique based on the Bayes theorem. It is one of useful supervised learning algorithm. Naive Bayes is “naıve" because it makes the assumption that features are independent from each other. This is Naive because it almost never true. Naive Bayes is a highly intuitive classification algorithm. It asks the question, “In view of these features, does this measure fall into class A or B?” to all data points and answers by taking the proportion of all preceding measurements with the same characteristics belonging to class A multiplied by the proportion of all measurements of class A. If this number is higher than the corresponding calculation for class B, then we can say that the measure belongs to class A (Fig. 5).

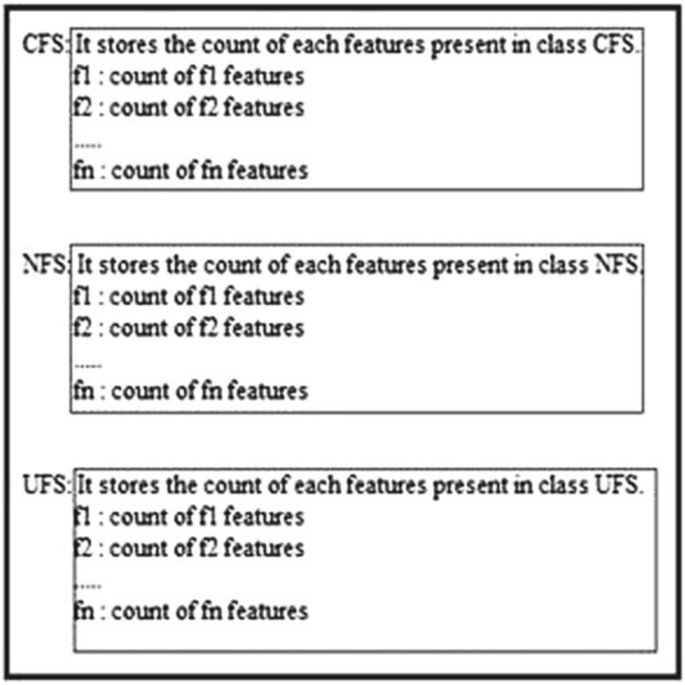


Fig. 5 Frequency of word corresponding to each class

4 Results Simulation

Naïve Bayes classifiers are a popular choice for classification problem because it is easy to write; classifier model take less time to build, and the model can be changed using new instruction data without having to rebuild the model and many more. In this paper, we developed the new version of Naïve Bayes classifier without making any assumptions about the independence of the features. A major step in this algorithm is to add edges between the characteristics that capture correlation between them. We have presented the result with Claim Buster dataset which is available aai.org platform. We applied our method to real-world text classification problem and showed that it worked better if we have smaller data corresponding to one or more classes. After, implementing Naive Bayes as mentioned in the paper, we are able to achieve a significant result which is totally able to classify classes to much more extend. From the above classification metric, we can easily see in Table 1 that after doing all the modification in Naive Bayes classifier, we are able to achieve the precision of about 93%. We also calculate the precision of our dataset with various algorithms like Naive Bayes, random forest classifier, and support vector machine which are present in sklearn library and compare it with our improved Naive Bayes classifier.

Table 1 Precision values for different classifiers

CLASS	Improved NBC	NBC	RFC	SVM
NFS	0.93	0.799	0.722	0.873
UFS	0.52	0	0.358	0.43
CFS	0.81	0.805	0.701	0.724

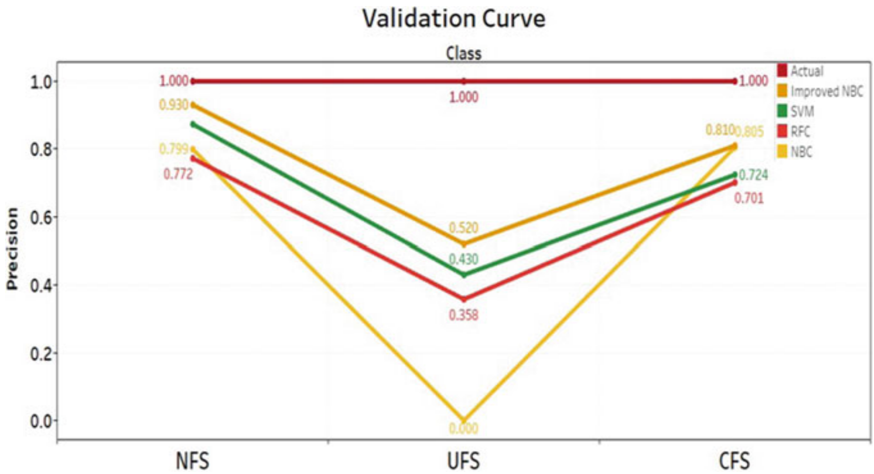


Fig. 6 Precision between different models

As we have seen previously to calculate the probability, we would have to make a lot of multiplication (as many times as the words in paragraph). And the problem is that all the probability numbers will be under one, and in general, it will be very small. As everyone knows, the product of two small numbers $0 < a, b < 1$ will end up being a smaller number (Fig. 6).

5 Conclusion and Future Scope

Herein, we have implemented an improved Naive Bayes classifier to classify text into their categories. We show that, essentially, the dependence distribution, i.e., how the local dependence of a node distributes in each class, evenly or unevenly and how the local dependencies of all nodes work together, consistently (supporting a certain classification) or inconsistently (canceling each other out), plays a crucial role. Our technique of classifying text, hence classify new samples, with the help of the pattern learned by it in the training dataset. The objective is to show you improved the Naive Bayes classifier. We worked on the assumption made by the Naïve Bayes classifier, zero-handled probability, or fast calculating the testing data with accurate probability function. So, by using the improved Naive Bayes classifier, our paper is now able to classify the sentences whether it is checked worthyfactual statement, non-factual statement, and unimportant factual statement.

References

1. Kaviani P, Dhotre S, Classification of Naive Bayes Classifier. Int J Adv Eng Res Dev
2. Geiger D, Friedman N, Bayesian network classifiers. SRI International, 333 Ravenswood Ave., Menlo Park, CA 94025
3. Rish, An empirical study of Naive Bayes classifier. T.J. Watson Research Center
4. Stephens CR, Ruiz Linares A, When is the Naive Bayes approximation not so Naïve. C3 Center for Complexity Sciences, National University of Mexico
5. Ikonomakis M, Kotsiantis S, Text classification using machine learning techniques. University of Patras
6. Gencheva P, Nakov P, A context-aware approach for detecting worth-checking claims in political debates. Sofia University "St. KlimentOhridski", Bulgaria Qatar Computing Research Institute, HBKU, Qatar
7. Kowsari K, Meimandi KJ, Text classification algorithms. University of Virginia, Charlottesville, USA
8. Hassan N, Arslan F, Toward automated fact-checking: detecting check-worthy factual claims by claim buster. University of Texas at Arlington
9. Tabassum A, Patil RR, Text pre-processing feature extraction techniques in natural language processing. GSSSIETW, Mysore
10. Vidhya S, Asir Antony Gnana Singh D, Feature extraction for document classification. Int J Innov Res Sci Eng Technol

A Machine Learning Framework for Document Classification by Topic Recognition Using Latent Dirichlet Allocation and Domain Knowledge

B. Lavanya and U. Vageeswari

Abstract The quantity of unstructured text data in digital archive is continually expanding due to the exponential growth of information technology, so the tasks of analysing, organizing, classifying, and summarizing text have become a big challenge. Since the manual classification of text documents requires a lot of human resources, finance, and time, automatic text classification is obligatory. Latent Dirichlet allocation (LDA) is an unsupervised machine learning algorithm often used in topic modelling. The output of the topic modelling algorithm can be used logically to classify documents. The LDA model is plagued with domain-specific terms. A novel latent Dirichlet allocation (LDA) with domain knowledge framework for document classification was introduced. The experiment was carried out using a dataset with five different categories of data. The experiments showed that LDA with domain knowledge gives better results than standard LDA and LDA using the TF-IDF model. Precision, Recall, $F1$-score, accuracy, and Purity were all improved using the proposed framework.

1 Introduction

The quantity of unstructured text data in digital archive is continually expanding due to the exponential growth of information technology, so the tasks of analysing, organizing, classifying, and summarizing text [1] have become a big challenge. Text mining is the task of extracting needful information from the unstructured text. The process of structuring the input text, identifying patterns within the structured data, and finally evaluating and interpreting the output is all covered by text mining. Text categorization, text clustering, and document classification are all common text mining tasks. Among the techniques used include summarization, keyword extraction, sentiment analysis, entity-relation modelling, concept/entity extraction, and so on.

B. Lavanya (✉) · U. Vageeswari
Department of Computer Science, University of Madras, Chennai, India
e-mail: lavanmu@gmail.com

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_42

Because manual text categorization consumes a significant amount of resources, money, and time, automatic text categorization is required. Text categorization uses machine learning or other technologies to categorize documents automatically. Text classification [2] can be performed in any of the following levels:

Document level: The entire document is treated as a single unit in the classification model.
Paragraph level: Each paragraph in a text is treated as a distinct unit in the classification model.
Sentence level: The classification model treats each sentence in a paragraph as an independent entity.
Sub-sentence level: The classification model treats a sentence's sub-expression as a single unit.

Topic models (TM) are generative models used in machine learning and natural language processing. They provide a probabilistic framework [3]. The output of the topic modelling framework can be used in many areas of NLP and information retrieval for classification or clustering of text, dimensionality reduction, and recommendation systems. Supervised, unsupervised, and semi-supervised TM methods are the three types of TM methods. It can use structured or unstructured data. Some of the application domains of TM are health care, agriculture, education, social-network opinion analysis [4], finance, recommendation system, manufacturing, bioinformatics, and etc. There are many topic modelling methods that are commonly used in text mining, namely latent Dirichlet allocation (LDA) [5], latent semantic analysis (LSA) [6], non-negative matrix factorization (NMF) [7], principal component analysis (PCA) [8], and random projection (RP) [9]. Among these LDA is mostly used TM method [10].

LDA is an unsupervised machine learning algorithm often used in topic modelling. LDA output can be used logically to classify the documents according to the topic it belongs to [11]. The LDA model, on the other hand, is plagued with domain-specific terms. Words that are unique to a domain are given extra weight in the word list for all topics. This must be addressed, and domain-specific terms that contribute more to the total word list must be identified.

A machine learning framework for document classification by topic recognition using latent Dirichlet allocation and domain knowledge was introduced. The framework finds domain-specific keywords that contribute the most to the word list for all topics, eliminates them from the LDA dictionary, and distributes their weight among other words in the topic. Results are compared with standard LDA and LDA with TF-IDF models. The proposed framework outperforms the current models in terms of outcomes.

2 Related Work

The latent Dirichlet allocation method is combined with the knowledge gathered from word embeddings in this hybrid technique [12]. Word2Vec extends the meaning of words in topics. The latent Dirichlet allocation is used initially in the hybrid approach (LDA). The algorithm's second phase incorporates the Word2Vec-acquired model words. The goal is to expand each topic by using terms that are comparable to the most typical words for that topic. The hybrid method was evaluated on four datasets and found to be superior on three of them.

The LDA approach for topic text categorization adds a topic-category distribution variable to LDA and creates the most pertaining category [13]. Documents in the model are divided into categories, with each category having its own collection of "themes". Each document is created in the category to which it is most likely to belong by identifying the category to which it is most related. It eliminates incorrect topic-word assignment by restricting the generating scope through the topic-category distribution parameter.

An enhanced short text classification approach was proposed based on the LDA topic model as well as the K-nearest neighbour algorithm [14]. The probabilistic themes that are generated help to make the texts better semantically coherent while lowering the text's sparsity. It also provides a new topic similarity metric based on the specific topic matrix and the relationship between two short texts' discriminative words. A small text dataset for experiment validation is built by browsing the Sina News portal for entries. A probabilistic topic model-based solution was presented, as well as a new method for assessing short text similarity.

By considering the document as a collection of word embeddings and subjects as multivariate Gaussian distributions in the input space, a novel approach for topic modelling is presented [15]. It looks at several collapsed Gibbs sampling approaches and develops scalable algorithms that improve the naïve implementation asymptotically.

3 Methods

3.1 Text Preprocessing

For each data mining activity, preprocessing is the initial step. It is a very vital step for text mining since the text is unstructured in nature. The basic preprocessing steps for text are normalization, tokenization, stemming, and lemmatization. Normalization is the process of converting the text in uniform format by removing punctuation, removing unwanted tags, and extracting text from XML and HTML files. Splitting the text into smaller parts is known as tokenization. Tokens are words. Converting the word into its non-changing stem word is stemming (e.g. processing into process).

Finding the root word is lemmatization (e.g. better into good). Choosing the necessary preprocessing step depends on the application and algorithm.

3.2 Text Vectorization

Text vectorization is the process that converts text into numerical form. There are various vectorization methods for text. Bag of words (BoW) term frequency, binary term frequency, normalized term frequency, and normalized TF-IDF are the most often utilized vectorization approaches [16]. Binary term frequency identifies whether a term is present (1) or absent (0) in a document. Bag of words (BoW) term frequency detects the frequency of terms in the document. Normalized term frequency finds the document's normalized BoW term frequency. Term frequency-inverse document frequency (TF-IDF) is used to identify normalised TF-IDF in documents. Among these BoW and TF-IDF are most widely used for the LDA algorithms.

3.3 BoW

Bag of words representation is the primary technique suggested by information retrieval scientists to represent text corpora; it is a simple method for converting unstructured text to structured data word by word. The grammar and sequence of occurrence of the term in the corpus are ignored by the BoW model. It just looks at a few instances of the term in the corpus. The BoW model has a sparse representation.

3.4 Term frequency-inverse document frequency (TF-IDF)

TF: Term frequency is a statistic that measures how frequently a term appears in a piece of writing. It is standardized by dividing the total number of words in a document by the number of terms in the document. Equation 1 shows how to calculate the TF Value.

$$\mathrm{tf}(t, d) = f_{t,d}/w \tag{1}$$

IDF: Inverse document frequency (IDF) is a statistic for assessing the significance of a sentence. It scales up the unusual words, while scaling down the frequent ones by computing the logarithm of the number of documents in the corpus divided by the number of documents where the given word appears. IDF value calculation is shown in Eq. 2.

$$\mathrm{idf}\,(t) = \log(N/(\mathrm{d}f + 1)) \tag{2}$$

TF-IDF: stands for term frequency-inverse document frequency. It is a numerical statistic that measures the significance of a word in a collection or corpus of documents. The significance of a word increases in proportion to how many times it appears in the text, but this is counterbalanced by the term's frequency in the corpus [17]. Equation 3 is used to calculate the TF-IDF value.

$$\text{tfidf}(t, d, D) = \text{tf}(t, d) * \text{idf}(t, D) \tag{3}$$

3.5 LDA

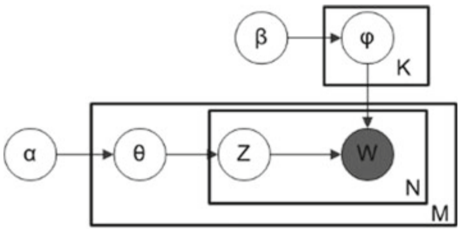
Latent Dirichlet allocation is an unsupervised algorithm that assigns each document a value for each defined topic. Latent means have hidden. Dirichlet is a type of probability distribution which is first introduced by Blei et al. [5]. Each document is treated as a collection of topics, and each topic is treated as a collection of words using LDA. For the LDA model, the input is a vectorized document. Figure 1 shows the plate notation of the LDA topic model

The boxes are "plates" that represent replicates, or things that repeat themselves. The outer plate shows articles, while the inner plate indicates the document's repeated word locations, each of which is linked to a certain topic and choice of words.

M denotes the number of documents
N_i is number of words in a given document (document i has N_i words)
α is the parameter of the Dirichlet prior on the per-document topic distributions
β is the parameter of the Dirichlet prior on the per-topic-word distribution
K denotes number of topics
θ_i is the topic distribution for document i
ϕ_k is the word distribution for topic k
z_{ij} is the topic for the jth word in document i
W_{ij} is the specific word.

According to LDA, each document may be characterized as a probability distribution over latent themes, with a common Dirichlet prior across all texts. Each latent topic is modelled as a probabilistic distribution across words in the LDA model, with

Fig. 1 LDA plate notation



topic-word distributions sharing a common Dirichlet prior. LDA models D according to the following generating process given a corpus D consisting of M documents, each document d comprising N_d words ($d \in 1, ..., M$).

(a) From a Dirichlet distribution with parameter, choose a multinomial distribution t for topic t ($t \in 1, ..., T$).
(b) From a Dirichlet distribution with parameter, select a multinomial distribution d for document d ($d \in 1, ..., M$).
(c) In document d, for a word w_n ($n \in 1, ..., N_d$),

i From θ_d, choose a topic z_n.
ii From $\varphi_z n$ choose a wordw_n. Words in documents are merely seen variables in the above generating process, whereas other variables (θ and φ) and hyper parameters (α and β) are latent variables. W is the only observable variables, and the other variables are latent variables. The probability of observed data D for a corpus is calculated and derived using Eq. 4.

$$p(D|\alpha, \beta) = \prod_{d=1}^{M} \int p(\theta_d|\alpha) \left(\prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn}|\theta_d) p(w_{dn}|z_{dn}), \beta \right) d\theta_d \tag{4}$$

The parameters of the topic Dirichlet prior α have been defined, as well as the distribution of words over topics, which has been drawn from the Dirichlet distribution β. The number of topics is T, the number of documents is M, and the vocabulary size is N. For corpus-level topic distributions, the Dirichlet-multinomial pair (α, θ) is used. For topic-word distributions, the Dirichlet-multinomial pair (β, φ) is given. The variable θ_d is variable at the document level, sampled once per document. The variables z_{dn} and w_{dn} are word-level variables that are sampled once for each word in each text document.

4 LDA with domain knowledge

The LDA with domain knowledge framework removes the domain-specific words which influence the LDA model more than other topic words. The general stopword removal will not remove domain-specific words. And more importantly, all domain-specific words should not be removed, only the words that occur in all topics should be ignored in topic identification. The weightage of the domain-specific word is now shared among other words. The output of the model is a fuzzy set. The topic with high weightage is considered as the topic of the document.

Figure 2 shows the LDA with domain knowledge framework. The corpus is preprocessed as the initial step. Lower case conversion, tokenization, stopword removal, stemming, and lemmatization are the preprocessing steps for this framework. The text should then be converted to vectors. BoW method text vectorization is done. The

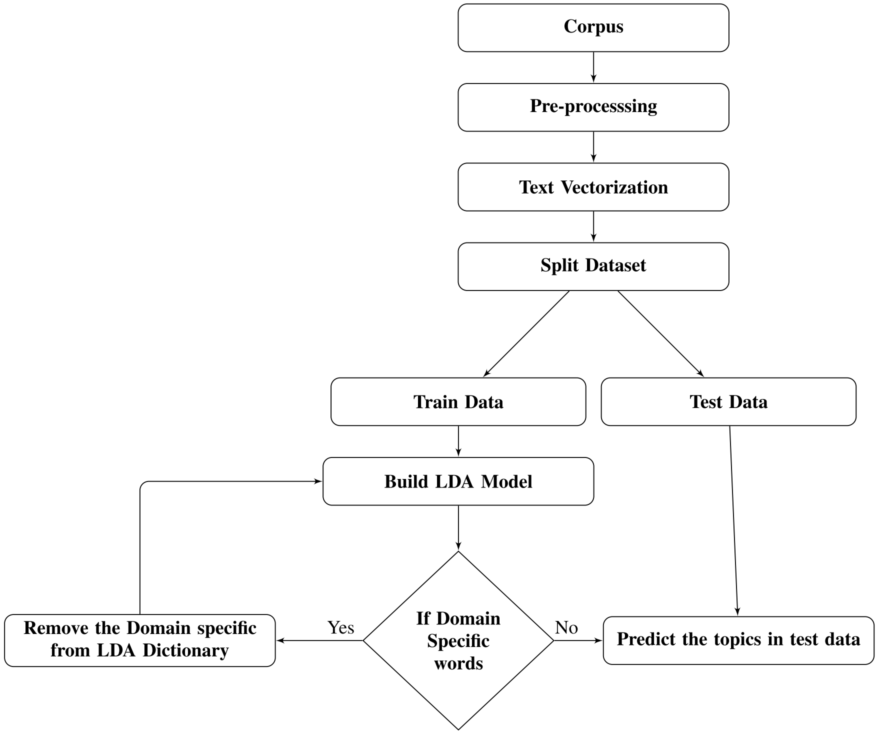


Fig. 2 LDA with domain knowledge

data is then divided into train and test groups. Train data was used to construct the LDA model. The suggested technique is used to check the top n words on each subject word list for domain-specific keywords. Apply the model to forecast test data topics if no domain-specific keywords exist. Before re-building the LDA model, remove any domain-specific words from the LDA lexicon. This approach detects and eliminates domain-specific words from the LDA lexicon that have a substantial influence on topic allocation.

The algorithm demonstrates how to locate domain-specific terms. The algorithm receives a collection of topic-word lists from the LDA model as input. The algorithm looks for terms that exist in all of the subject word lists and contribute more to the overall probability of the topic. The loop iterates through all topic-word lists and finds the domain-specific words that contribute much in weightage.

Algorithm 1: DST:Domain-specific Terms

```
Input: TWL
/* TWL - Topics_Word_list                                        */
Output: DSW
/* DSW - Domain_Specific_Word                                    */
Function Find_domain_specific_word(TWL)
    TOL ← TWL[0]
    // TOL - Topicoverlap
    DSW ← Ø
    foreach WL in TWL do
        DSW ← WL ∩ TOL
        TOL ← DSW
    return DSW
```

5 Experimental Results

5.1 Data Set

We chose BBC News dataset for our experiment, which comprises five areas of data: sport, technology, business, politics, and entertainment. Figure 3 shows document distribution among topics. It has 2225 news documents. The count of documents in each category is tabulated in Table 1.

Fig. 3 Document distribution among topics

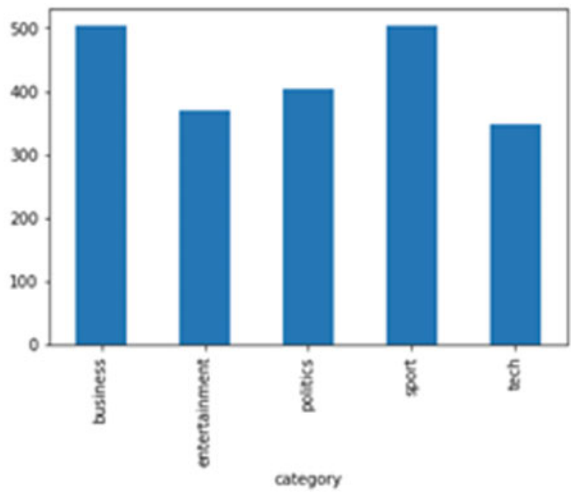


Table 1 Count of documents in each topic

Category	Number of documents
Business	510
Entertainment	386
Politics	417
Sport	511
Technology	401

5.2 *Result Discussion*

We compared our framework with standard LDA and LDA with TF-IDF as the text vectorization method. A word's relevance to a document in a corpus or collection is represented by the TF-IDF. Table 2 shows the top five words in all topics from the standard LDA Model in the BBC News dataset. The bold words are domain-specific words which influence the model more and gives prejudiced result, and it is found by the algorithm. Table 3 shows the top five words in all topics from the LDA with domain knowledge model BBC News dataset.

Table 4 shows the topic-word list weightage in the standard LDA Model. Table 5 shows the topic-word list weightage in LDA with Domain Knowledge model. The topic-word list plays a vital role in topic modelling. Figure 4 shows the word count and word weightage in each model for the top 5 words in each topic. Each keyword's word counts and weights are shown on the same graph. The chart shows the improvement in weight allocation for keywords than domain-specific words.

Table 2 Top five topic words in standard LDA

Topic 0	Topic 1	Topic 2	Topic 3	Topic 4
Say	**Say**	**Say**	Game	**Say**
Govern	**Year**	**Year**	**Say**	Company
Labour	People	Best	England	**Year**
Elect	Mobile	Play	Play	Firm
Year	Service	Film	**Year**	Sale

Bold words are domain specific words

Table 3 Top five topic word in LDA with domain knowledge

Topic 0	Topic 1	Topic 2	Topic 3	Topic 4
Labour	Company	Govern	Play	Film
Elect	Market	Country	Time	People
Party	Rise	People	Player	Music
People	Firm	World	England	Best
Blair	Bank	Nation	Go	Mobile

Table 4 Standard LDA topic-word weightage

Topic 0	0.027 * "Say" + 0.008 * "People" + 0.008 * "Govern" + 0.005 * "Minist" + 0.005 * "Elect"
Topic 1	0.017 * "Say" + 0.009 * "Game" + 0.007 * "Year" + 0.007 * "Music" + 0.006 * "People"
Topic 2	0.012 * "Film" + 0.010 * "Say" + 0.010 * "Year" + 0.009 * "Best" + 0.007 * "Play"
Topic 3	0.017 * "Say" + 0.006 * "Club" + 0.006 * "Game" + 0.006 * "Play" + 0.005 * "Time"
Topic 4	0.022 * "Say" + 0.013 * "Year" + 0.008 * "Market" + 0.006 * "Rise" + 0.005 * "Price"

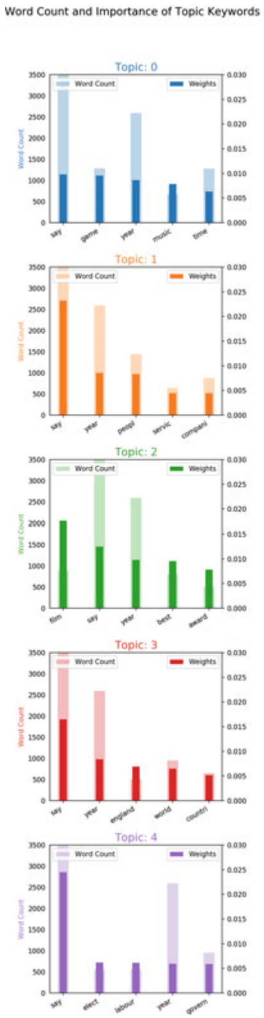
Table 5 LDA with domain knowledge topic-word weightage

Topic 0	0.008 * "Play" + 0.007 * "England" + 0.006 * "Time" + 0.005 * "Best" + 0.005 * "Player"
Topic 1	0.006 * "Rise" + 0.006 * "Market" + 0.006 * "World" + 0.005 * "Month" + 0.005 * "Price"
Topic 2	0.008 * "Govern" + 0.008 * "People" + 0.007 * "Elect" + 0.007 * "Labour" + 0.007 * "Party"
Topic 3	0.014 * "Film" + 0.006 * "People" + 0.006 * "Music" + 0.005 * "Best" + 0.005 * "Star"
Topic 4	0.007 * "Company" + 0.007 * "Firm" + 0.006 * "Phone" + 0.006 * "Mobile" + 0.005 * "People"

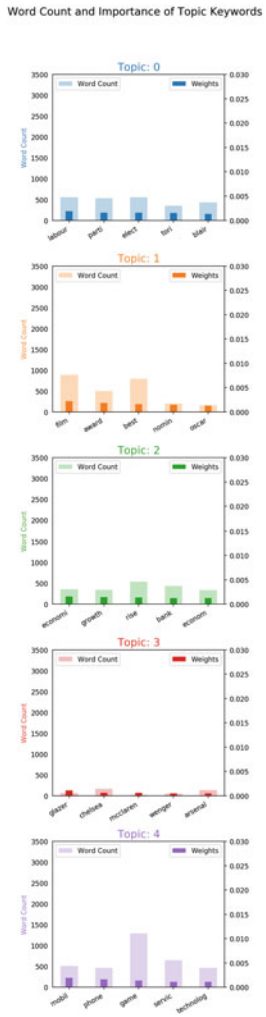
The LDA with domain knowledge model is assessed by precision, recall, and $F1$-score. Values for accuracy and purity are computed and reported. All models' accuracy, $F1$-score, precision, recall, and purity value are included in Table 6. The experiments showed that LDA with domain knowledge gives better results than standard LDA and LDA using the TF-IDF model. The LDA with domain knowledge framework increased precision by 18%, recall by 20%, $F1$-score by 21%, accuracy by 19%, and purity by 14%.

6 Conclusion and Future Work

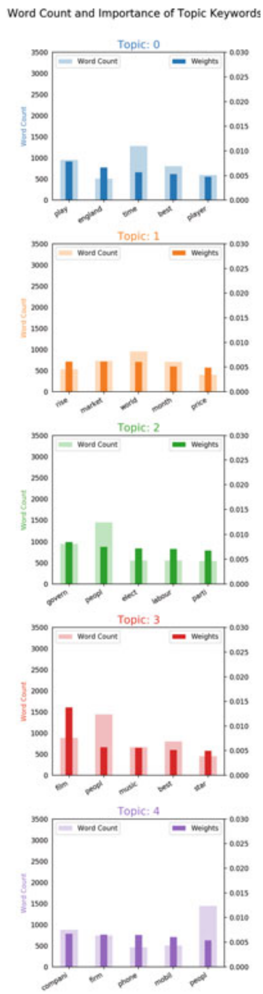
In-text mining, topic models play a significant role. There are many text classification models in computer science. We have chosen LDA since it is a probability-based method. LDA requires a number of topics(K) as one of the inputs. The proposed LDA with domain knowledge framework gives better results than standard LDA and LDA using TF-IDF by removing domain-specific words. Here, in this experiment from the top five words, the dominant words among topics are removed. Depending on the amount of the sample and the range of topics, it may be altered to the ten leading or more M words. The LDA with domain knowledge framework increased



(a) LDA



(b) LDA with TF-IDF



(c) LDA with Domain Knowledge Model

Fig. 4 Word count and word importance in three models

Table 6 Precision, recall, $F1$-score values, accuracy, and purity

Model	Precision	Recall	$F1$-score	Accuracy	Purity
Standard LDA	0.53199	0.51173	0.49935	0.52816	0.57746
LDA with TF-IDF	0.53362	0.53755	0.44874	0.53755	0.53755
LDA and domain knowledge	**0.71162**	**0.71596**	**0.70610**	**0.71596**	**0.71596**

The numbers specified in bold are result from proposed method, that achieved better result than the existing methods precision by 18%, recall by 20%, F1-score by 21%, accuracy by 19%, and purity by 14%. It can be further improved by introducing wordnet as one of the components. Part of speech (PoS) can also be included as one of the components in future work.

Acknowledgement This work is supported by University Grants Commission, under Junior Research Fellowship (JRF).

References

1. Lavanya B, Vageeswari U (2021) An assortment of query based summarization technique (QBS)—a study, pp 659–666
2. Kowsari K, Jafari Meimandi K, Heidarysafa M, Mendu S, Barnes L, Brown D (2019) Text classification algorithms: a survey. Information 10(4):150
3. Kim S-B, Han K-S, Rim H-C, Myaeng SH (2006) Some effective techniques for naive bayes text classification. IEEE Trans Knowl Data Eng 18(11):1457–1466
4. Aborisade O, Anwar M (2018) Classification for authorship of tweets by comparing logistic regression and naive bayes classifiers. In: 2018 IEEE international conference on information reuse and integration (IRI). IEEE, pp 269–276
5. Blei DM, Ng AY, Jordan MI (2003) Latent dirichlet allocation. J Mach Learn Res 3:993–1022
6. Deerwester S, Dumais ST, Furnas GW, Landauer TK, Harshman R (1990) Indexing by latent semantic analysis. J Am Soc Inf Sci 41(6):391–407
7. Berry MW, Browne M (2005) Email surveillance using non-negative matrix factorization. Comput Math Organ Theor 11(3):249–264
8. Jolliffe IT (2002) Springer series in statistics. Principal component analysis, vol 29
9. Wang X, McCallum A (2006) Topics over time: a non-markov continuous-time model of topical trends. In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, pp 424–433
10. Gopal GN, Kovoor BC, Mini U (2021) Keyword template based semi-supervised topic modelling in tweets. In: International conference on innovative computing and communications. Springer, pp 659–666
11. Pavlinek M, Podgorelec V (2017) Text classification method based on self-training and lda topic models. Expert Syst Appl 80:83–93
12. Jedrzejowicz J, Zakrzewska M (2020) Text classification using lda-w2v hybrid algorithm. In: Intelligent decision technologies 2019. Springer, pp 227–237
13. Zhao D, He J, Liu J (2014) An improved lda algorithm for text classification. In: 2014 international conference on information science, electronics and electrical engineering, vol 1. IEEE, pp 217–221
14. Chen Q, Yao L, Yang J (2016) Short text classification based on lda topic model. In: 2016 international conference on audio, language and image processing (ICALIP). IEEE, pp 749–753
15. Das R, Zaheer M, Dyer C (2015) Gaussian lda for topic models with word embeddings. In: Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing (volume 1: long papers), pp 795–804
16. Robertson S (2004) Understanding inverse document frequency: on theoretical arguments for IDF. J Document
17. Jones KS (1972) A statistical interpretation of term specificity and its application in retrieval. J Document

Secure AI-Based Flying Ad Hoc Networks: Trusted Communication

Sadoon Hussein, Abida Thasin, Ahmed Sami, and A. Sabitha Banu

Abstract Unmanned aerial vehicles are real-time applications for flying networks which encourage multi-UAV structures. A multi-UAV system has a corporate behavior and can finish a mission efficiently. Coverage issues between ground stations and aerial vehicles are easily solved using FANETs. Because of the dynamic nature of flying vehicles, there are several security concerns. Routing protocols can upgrade and secure communication channels in flying networks. In order to increase the lifeline of aerial networks, a qualified routing research study is also carried out. According to simulation findings, zone routing protocol has maintained to secure communication channels and promises higher security without incurring computational expense.

Keywords FANETS · UAV · DSR & ML

1 Introduction

Flying ad hoc networks is having collection of UAVs. Where, the use of aerial vehicles in today's society has changed people's lives, especially after COVID-19. Flying vehicles operate in a self-organized network environment. However, in the field of aerial networks, a new dimension known as data networking has emerged. The proposed system monitors based on trust to maintain data security inside the network

S. Hussein · A. Thasin
Department of Physic, College of Science, University of Mosul, Mosul, Iraq
e-mail: sadosbio113@uomosul.edu.iq

A. Sami
Department of Computer Science, College of Computer Science and Mathematics, University of Mosul, Mosul, Iraq
e-mail: ahmed.s.nori@uomosul.edu.iq

A. Sabitha Banu (✉)
Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, Tamil Nadu, India
e-mail: 16phcsf003@avinuty.ac.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_43

in order to avoid the ström of data packets. Cloud-based aerial networks have been considered the main technology behind content-centric networking and named data networking for flying vehicles [1]. The need for effective and secure communication routing techniques has grown in response to the dynamic mobility of flying vehicles. Routing protocols create optimal path from source to destination; therefore, maintaining routing table's table-driven algorithms can be utilized. However, for optimization, Hopfield neural networks are merged with the traditional routing method called dynamic source routing. Continuous Hopfield neural networks is compared with traditional scheme DSR in terms of average end-to-end delay, throughput, and packet delivery ratio. The novel CHNN-DSR is artificial intelligence-based routing approach which shows better results in contrast with DSR [2].

As in aerial networks, the major issue is energy efficiency/power which has direct effect on the performance of flying vehicles. For this purpose, I Khan et al. introduced a hybrid protocol with a basic principle of ant colony optimization while state-of-the-art parameter is initiated and merged with the technique due to that algorithm is named E-AntHocNet [3]. In between the aerial networks and base station, wireless communication technology is utilized to enhance the capabilities of UAV-based networks. To improve the communication, wireless channels must be upgraded with the artificial intelligence subject called decision tree. Researchers have proposed a novel technique to find out the actual location, optimize receive signal strength, and estimate distance from one node to another. While the experimentation is performed in two scenarios which include 2D and 3D [4]. The study of artificial intelligence-based routing techniques has improved flying networks. In addition, routing plays important role in connecting one node with another. Also UAV's are used to overcome on delay factor in flying vehicles [5]. Reinforce-based learning using routing techniques is deployed in the area of Internet of flying Vehicles which improves packet drop rate, bandwidth utilization, network throughput, and end-to-end delay. Random way point mobility pattern is used in the study to enhance the pauses/delay in communication. Also this research study have a detailed analysis on wireless communication technologies [6].

In the last few decades, sports and health industries have become advanced in terms of technology. The status of athletes can be monitored using aerial vehicles to give accurate information which will reduce injuries in players. For better communication in flying networks, artificial intelligence-based routing scheme is introduced called AntHocNet which helps in decision-making. Boundless area mobility model is incorporated to evaluate network performance with different metrics [7]. Important challenge for Internet of Networks are security threats. As the IoT-based networks face a lot of vulnerabilities through these loop holes intruder use different attacks like denial of service and ping of death. So to enhance the security and detect attacks easily, a novel IDS is designed which use to consider the probability of missed detection. In addition, a threshold is incorporated to minimize false alarm and a decision rule which depends on the number of over length data packets [8]. Wang et al., on the other hand, addressed four major features of 6G networks: intelligent edge computing, artificial intelligence, radio communication, and 3D intercoms, all of which provide solutions to privacy and security challenges [9]. As the automobile industry progresses technologically, the Internet of Vehicles will communicate with the surrounding devices on the ground. As a result, machine learning is one of the most effective tools for resolving decision-making in communication channels [10]. Due to the ever changing structure of aerial networks, the ability to establish line of sight using next generation wireless communication. AI merger with flying vehicles characterize and enhance battery timing, memory consumption, and solve congestion issues to secure communication in UAV networks [11].

2 Literature Survey

Artificial intelligence-enabled aerial networks are a new paradigm to improve the dynamics of flying vehicles. While deploying multiple UAV's, which will collect information from the positioned sensor nodes. Moreover, AI technology will boost swarm UAVs to find out optimal solution. This paper focus to design resource distribution in aerial vehicles which utilize artificial intelligence to enhance performance and upgrade adaptability [12]. In flying vehicles, searching possible routes is a major problem. For this purpose, genetic algorithm-based routing protocol is formulated which helps to optimize routes, throughput, minimize delay, and stabilize the network [13]. In multi-UAV structure, reliable communication needs protocols which must efficient, scalable, and accurately adapt the dynamic nature of aerial networks [14]. Due to the dynamic pattern of aerial vehicles from ground, the Internet of Everything will play a better part to collect data from the environment. Masood Ahmad et al. proposed a protocol which is having the attributes of onlooker bees and IoE are placed using Poisson distribution to increase the delivery ratio and life-time. Also unmanned aerial vehicles are integrated with IoE to reduce end-to-end delay in the network [15]. To overcome on the insatiability of UAV ad hoc networks, a novel protocol is designed to establish better communication links in between network. However, EV-AODV prolongs network lifetime and gives higher packet delivery ratio which easily stabilizes the communication in flying ad hoc networks [16]. Some authors contributed in ad hoc networks for better communication. Where Table 1 describes information regarding protocols and aerial ad hoc networks. Also security-based routing protocols are shown in Fig. 1 which can be used to secure flying networks.

3 Proposed Algorithm

Zone routing protocol (ZRP) is utilized as proposed solution which is a hybrid approach. ZRP is having the features of both reactive and proactive behavior. In this algorithm, the deployed flying vehicles are divided in zones which must have clust head.

Table 1 Ad hoc networks and routing protocols

Refs no./author	Ad hoc network/drone	Protocol information
Sang [17]	FANET	Ground control system-based routing (GCS)
Ibrahim [18]	UAV	OLSR routing protocol
Mairaj [19]	Mobile ad hoc network	UAVNet
Kumar [20]	Aerial	Neuro-fuzzy interference system
Shivahare [21]	MANET	Proactive (table driven) Reactive (on demand)(DSDV, DSR, and AODV)
Hu [22]	Ariadne	Dynamic source routing protocol
Doshi [23]	Wireless ad hoc networks	Dynamic source routing
Condomines [24]	UAV networks	Intrusion detection system (IDS),
Bautista [25]	FANET's	SrFTime and CRP
Sanzgiri [26]	Ad hoc network	ARAN
Darabkh [27]	UAV-to-UAV	Multidata rate mobility aware (MDRMA)
Iordanakis [28]	MANET	ARPAM
Khan [29]	FANETs	Topology-based routing
Mariyappan [30]	FANETS	AODV
Chaba [31]	MANETs	Ad hoc on demand distance vector (AODV) and dynamic source routing (DSR)

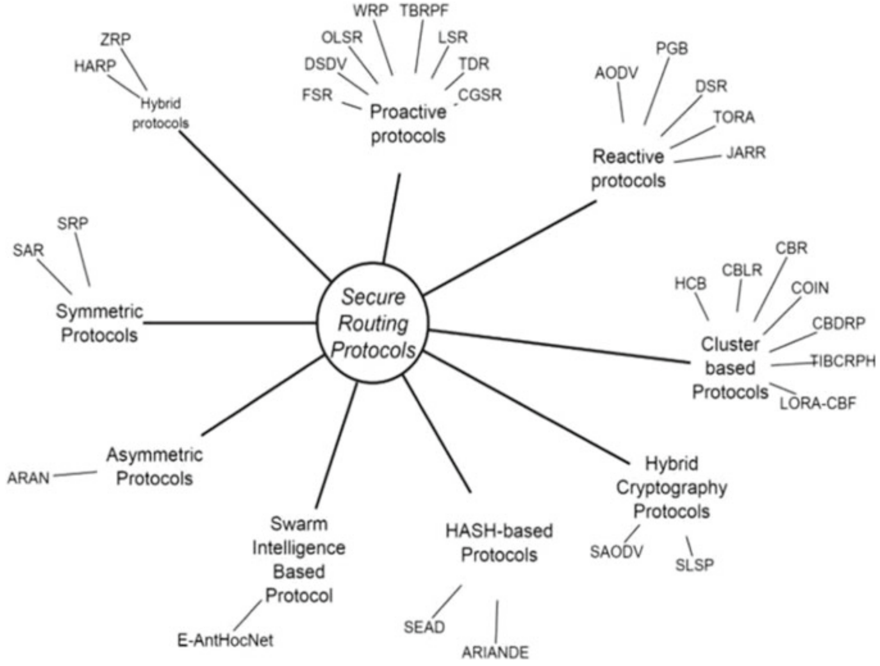


Fig. 1 Secure routing protocol

During inter-communication between network, as the intruder attack on some nodes which can affect the data packets while sending from source to destination. Apart from that type of artificial intelligence technique, ant colony optimization is utilized as AntHocNet in the network structure. Basically, the basic principle of AntHocNet is based on searching and finding food from nest to target which is inspired by biological evolution. However, zone routing protocol (ZRP) has shown better simulations results in terms of artificial intelligence-based AntHocNet.

4 Simulation Results

The experimentation is performed in network simulator-2. The basic topology consists of aerial vehicles which is trying to collect data from the ground sensor nodes and sends it to base station. For efficient communication, routing protocols which include AOMDV, DSDV, DSR, M-DART, ZRP, and AI-based AntHocNet is deployed in flying networks. Parameters like network throughput, network utilization, and packet delivery are used to evaluate network performance. Figure 1 shows to represent the behavior of routing techniques to secure communication. For this purpose, zone routing protocol and artificial intelligence-based AntHocNet give better results in comparison with other contemporary routing schemes. Table 2 utilizes the numerical representation of Fig. 1, Also Fig. 2 is the basic working model for zone routing protocol (ZRP) having different zones.

Table 3 describes network utilization process which is basically the proportion of the current system to give high rise to the amount of traffic that can be used. According to the mentioned numerical analysis in Table 3, zone routing protocol shows optimal results.

The most interesting metric is used to evaluate the performance of each routing protocol. Through this technique, the spoofed or attacked aerial vehicles in the network can be handled easily. Figure 3 illustrates that M-DART and AI-based AntHocNet show less packet delivery ratio. While zone routing protocol is considered better technique among the others.

Table 2 Throughput analysis

Throughput analysis	AntHocNet	AOMDV	DSDV	DSR	M-DART	ZRP
Minimum	24	98.8125	24	56	24	78.03125
Maximum	650.81257	701	742.25	712	226.875	659.125
Average	356.54868	598.4889	649.439	501.8935	83.20062	512.02
Standard deviation	160.34404	97.83563	82.2995	180.1105	41.40626	72.3446

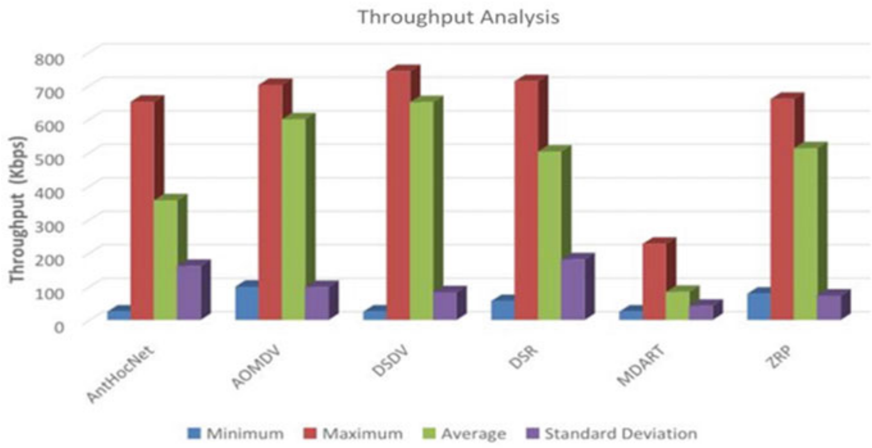


Fig. 2 Network throughput

Table 3 Network bandwidth/utilization analysis

Network utilization analysis	AntHocNet	AOMDV	DSDV	DSR	M-DART	ZRP
Minimum	720	1084.609	1028.625	1485.234	720	720
Maximum	8737.734	3492.766	3095.578	4772.984	3027.359	1320
Average	8737.734	3492.766	3095.578	4772.984	3027.359	1320
Standard deviation	698.7145	282.4205	261.3973	465.7308	334.9496	129.4262

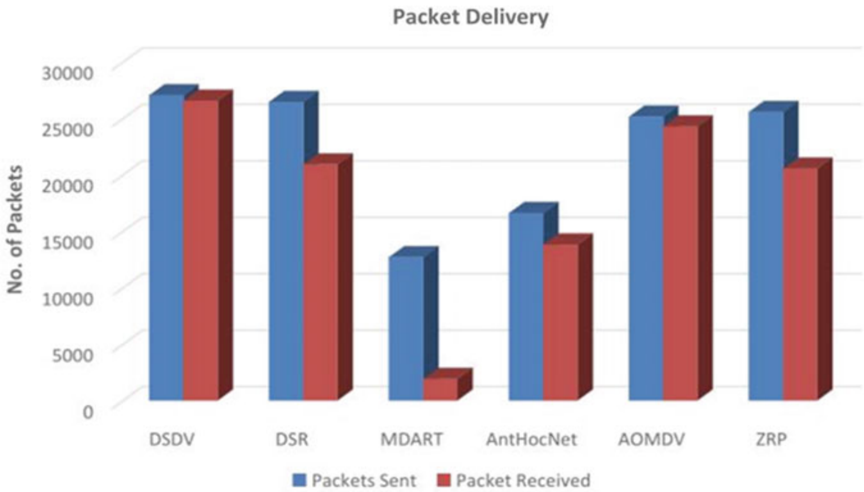


Fig. 3 Packet deliver

5 Conclusion & Future Challenges

AI-based routing protocol called AntHocNet works better in the pattern of aerial networks. The merger of AI with flying ad hoc networks will enhance to improve network lifetime. In the proposed technique, zone routing protocol, the concept of intruder is shown which directly attacked some flying nodes. As due to the flying nature of aerial vehicles, the installation of different routing algorithms in topology is used to secure communication. Zone routing protocol and AI-based AntHocNet have shown optimal results of metrics which include packet delivery, network utilization, throughput, and quality of experience. However, trying to secure more the communication channels in flying networks where machine learning, neural networks, and optimization techniques can used for future practices.

References

1. Barka E, Kerrache CA, Hussain R, Lagraa N, Lakas A, Bouk SH (2018) A trusted lightweight communication strategy for flying named data networking. Sensors 18(8):2683
2. Yang H, Liu Z (2019) An optimization routing protocol for FANETs. EURASIP J Wirel Commun Netw 2019(1):1–8
3. Khan IU, Qureshi IM, Aziz MA, Cheema TA, Shah SBH (2020) Smart IoT control-based nature inspired energy efficient routing protocol for flying ad hoc network (FANET). IEEE Access 8:56371–56378
4. Khan IU, Alturki R, Alyamani HJ, Ikram MA, Aziz MA, Hoang VT, Cheema TA (2021) RSSI-controlled long-range communication in secured IoT-enabled unmanned aerial vehicles. Mob Inf Syst
5. Khan IU, Shah SB, Wang L, Aziz MA, Stephan T, Kumar N (2021)Routing protocols & unmanned aerial vehicles autonomous localization in flying networks. Int J Commun Syst e4885
6. Khan IU, Nain Zukhraf SZ, Abdollahi A, Imran SA, Qureshi IM, Aziz MA, Hussian Shah SB (2020) Reinforce based optimization in wireless communication technologies and routing techniques using internet of flying vehicles. In: The 4th International conference on future networks and distributed systems (ICFNDS), pp 1–6
7. Khan IU, Hassan MA, Alshehri MD, Ikram MA, Alyamani HJ, Alturki R, Hoang VT (2021) Monitoring system-based flying IoT in public health and sports using ant-enabled energy-aware routing. J Healthc Eng 2021
8. Abdollahi A, Fathi M (2020) An intrusion detection system on ping of death attacks in IoT networks. Wirel Pers Commun, pp 1–14
9. Wang M, Zhu T, Zhang T, Zhang J, Yu S, Zhou W (2020) Security and privacy in 6G networks: New areas and new challenges. Digit Commun Netw 6(3):281–291
10. Ali ES, Hasan MK, Hassan R, Saeed RA, Hassan MB, Islam S, Nafi NS, Bevinakoppa S (2021) Machine learning technologies for secure vehicular communication in internet of vehicles: recent advances and applications. Secur Commun Netw 2021
11. Lahmeri MA, Kishk MA, Alouini MS (2021) Artificial intelligence for UAV-enabled wireless networks: a survey. IEEE Open J Commun Soc 2:1015–1040
12. Liu X, Chen M, Liu Y, Chen Y, Cui S, Hanzo L (2020) Artificial intelligence aided next-generation networks relying on UAVs. IEEE Wirel Commun 28(1):120–127
13. Wei X, Yang H, Huang W (2021) A genetic-algorithm-based optimization routing for FANETs. Front Neurorobot 15:81

14. Sang Q, Wu H, Xing L, Xie P (2020) Review and comparison of emerging routing protocols in flying ad hoc networks. Symmetry 12(6):971
15. Ahmad M, Ullah F, Wahid I, Khan A, Uddin MI, Alharbi A, Alosaimi W (2021) A bio-inspired routing optimization in UAV-enabled internet of everything. CMC-Comput Mater Continua 67(1):321–336
16. Wu J, Shi S, Liu Z, Gu X (2019) Optimization of AODV routing protocol in UAV ad hoc network. In: International conference on artificial intelligence for communications and networks. Springer, Cham, pp 472–78
17. Sang Q, Wu H, Xing L, Xie P (2020) Review and comparison of emerging routing protocols in flying ad hoc networks. Symmetry 12(6):971
18. Ibrahim MMS, Shanmugaraja P (2021) Optimized link state routing protocol performance in flying ad-hoc networks for various data rates of un manned aerial network. Mater Today Proc 37:3561–3568
19. Mairaj A, Baba AI, Javaid AY (2019) Application specific drone simulators: recent advances and challenges. Simul Model Pract Theory 94:100–117
20. Kumar K, Kumar S, Kaiwartya O, Kashyap PK, Lloret J, Song H (2020) Drone assisted flying ad-hoc networks: mobility and service-oriented modeling using neuro-fuzzy. Ad Hoc Netw 106:102242
21. Shivahare BD, Wahi C, Shivhare S (2012) Comparison of proactive and reactive routing protocols in mobile adhoc network using routing protocol property. Int J Emerg Technol Adv Eng 2(3):356–359
22. Hu YC, Perrig A, Johnson DB (2005) Ariadne: a secure on-demand routing protocol for ad hoc networks. Wireless Netw 11(1):21–38
23. Doshi S, Bhandare S, Brown TX (2002) An on-demand minimum energy routing protocol for a wireless ad hoc network. ACM SIGMOBILE Mob Comput Commun Rev 6(3):50–66
24. Condomines JP, Zhang R, Larrieu N (2019) Network intrusion detection system for UAV ad-hoc communication: from methodology design to real test validation. Ad Hoc Netw 90:101759
25. Bautista O, Akkaya K, Uluagac AS (2020) Customized novel routing metrics for wireless mesh-based swarm-of-drones applications. Internet of Things 11:100265
26. Sanzgiri K, Dahill B, Levine BN, Shields C, Belding-Royer EM (2002) A secure routing protocol for ad hoc networks. In: 10th IEEE international conference on network protocols, 2002. Proceedings. IEEE, pp 78–87
27. Darabkh KA, Alfawares MG, Althunibat S (2019) MDRMA: Multi-data rate mobility-aware AODV-based protocol for flying ad-hoc networks. Veh Commun 18:100163
28. Iordanakis M, Yannis D, Karras K, Bogdos G, Dilintas G, Amirfeiz M, Colangelo G, Baiotti S (2006) Ad-hoc routing protocol for aeronautical mobile ad-hoc networks. In: Fifth international symposium on communication systems, networks and digital signal processing (CSNDSP). Citeseer, pp 1–5
29. Khan MA, Khan IU, Safi A, Quershi IM (2018) Dynamic routing in flying ad-hoc networks using topology-based routing protocols. Drones 2(3):27
30. Mariyappan K, Christo MS, Khilar R (2021) Implementation of FANET energy efficient AODV routing protocols for flying ad hoc networks [FEEAODV]. Mater Today Proc
31. Chaba Y, Singh Y, Joon M (2010) Simulation based performance analysis of on-demand routing protocols in MANETs. In: Second international conference on computer modeling and simulation, pp 80–83

Biological Sequence Classification Using Deep Learning Architectures

Arrun Sivasubramanian, V. R. Prashanth, S. Sachin Kumar, and K. P. Soman

Abstract Finding similar biological sequences to categorize into respective families is an important task. The present works attempt to use machine learning-based approaches to find the family of a given sequence. The first task in this direction is to convert the sequences to vector representations and then train a model using a suitable machine learning architecture. The second task is to find which family the sequence belongs to. In this work, deep learning-based architectures are proposed to do the task. A comparative study on how effective various deep learning architectures for this problem is also discussed in this work.

Keywords Bio sequence · Perceptron · CNN · BiLSTM · GRU · SARS · SARS-CoV-2 · MERS

1 Introduction

The coronaviridae family belongs to the class of virus species that contain a single-stranded RNA. They can be further classified into four groups, i.e., α, β, γ, and δ coronavirus. Among these, the Middle East Respiratory Syndrome (MERS), the Severe Acute Respiratory Syndrome (SARS), and SARS-CoV-2 belong to the β coronaviridae family, but their subgenus are different: MERS and SARS under Merbecovirus and SARS-CoV-2 under Sarbecovirus subgenus. Thus, though these viruses are closely co-related, they differ in many aspects that include their effects

A. Sivasubramanian (✉) · V. R. Prashanth · S. Sachin Kumar · K. P. Soman
Centre for Computational Engineering and Networking, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, Tamil Nadu, India
e-mail: arrun.sivasubramanian@gmail.com

S. Sachin Kumar
e-mail: s_sachinkumar@cb.amrita.edu

K. P. Soman
e-mail: kp_soman@amrita.edu

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_44

on mankind when he/she is infected and the course for which medical treatment is required.

DNA sequences consist of four major bases. In RNA, Uracil replaces Thymine. Adenine pairs with Thymine and Guanine pairs with Cytosine using 2 and 3 weak hydrogen bonds, respectively. With the increasing power of computation in modern devices and advancements in deep learning, several attempts have been made to decipher the structural features seen in the genome of several disease-causing viruses. Typical sequence classification techniques rely on finding the common subsequences that predominantly occur in the species and trying to find out the properties that they possess. Needleman–Wunsch algorithm and Smith Waterman algorithm are some of the best dynamic programming approaches to find overlaps between nucleotide subsequences [1]. In this paper, we focus on finding one of the best methods to classify sequences of MERS, SARS, and SARS-CoV-2 using architectures among multilayer perceptron, convolutional neural nets, bidirectional long short memory, and gated recurrent unit, that help to identify essential characteristics and to classify the sequences [2, 3]. Section 2 deals with the related works done so far to classify sequences within the same family. Section 3 elaborates the methodology followed in our work and Sect. 4 describes the approach we propose, using the abovementioned architectures. Section 5 discusses the obtained results with a visual representation using ROC-AUC Curves and analysis of the sequences, while Sect. 6 discusses the appropriate model that can be used depending on the availability of resources and circumstances. Toward the end, at Sect. 7, we elaborate the future works that can be carried out and also give an overall gist of the paper.

2 Related Work

The efforts to represent nucleotides in genomes in decipherable ways is a quintessential task. The techniques include similarity in subsequences and probabilistic approaches [4, 5]. Our initial thought was, to express each nucleotide base as encodings, that will yield a n-digit binary number [5] using Ex-OR operation for capturing the patterns in various subsequences. Despite being a simple task, it could not capture patterns of forthcoming subsequences. Sequence classification of the coronaviridae family using support vector machines was done in [6] and certain other machine learning techniques to classify the sequences were done in [7] to get good results, but deep learning can capture non-linear relationships between subsequences more accurately. CNN-based hybrid models were explored in [8] to classify genomes of different families, but classifying subsequences of the same family is a challenging task. The second approach was to perform one-hot encoding of individual bases or the frequency information of k-mers or n-grams [9]. This approach has drawbacks, owing to its incapacity to capture the similarity relationships between n-grams. The BoW approach, as written in [10], does not consider the original index of k-mers in the subsequences. In [11], an approach of denoting sequences using the nucleotides and projecting them onto the Fermat spiral has been mentioned, but there will be many points in our case to project and obtain a good relation between nucleotides. With progress in the domain of Natural Language Processing, many discovered the co-relation between unigrams in nucleotide sequences with tokens in subsequences. This prompted the beginning of word embedding approaches for subsequence representations, purely performed based on mapping of nucleotides to numbers. However, the task implies that the quantitative magnitude of one nucleotide will be more than the other, which may bias the model and may not present the structural meaning. Yet, it simplifies the problem manifold and deep learning also helps in efficient classification of the sequences which are reflected in our results.

3 Methodology

In this section, we elaborately describe the functionality of each architecture used in the work. The different architectures we use are multilayer perceptron, convolutional neural network (CNN), bidirectional long short-term memory (BiLSTM) network, and gated recurrent unit (GRU). In these architectures too, the input can be of two different types. In case of MLP's, the input is either in the form of a vector of single nucleotides from the nucleotide sequences or in the form of k-mers that we obtain by splicing the same. The detailed architecture used and the obtained results have been discussed in subsequent sections. Perceptrons are used for supervised learning for classification in which an input (represented as a vector) belongs to a specific class. In general, perceptron consists of four parts: The input values, weights and bias, cumulative sum, and an activation function which calculates and scales the output in a linear or non-linear fashion. In this work, we use multilayer perceptron to make the architecture accommodate more non-linear relations easily.

CNNs belong to the sub-section of deep neural networks which classify and recognize certain characteristics from images and are used for examining and identification. It mainly denotes the mathematical function "convolution" which is a linear operation where a function is multiplied with a kernel to result in a third function that describes how the shape of one function is changed by another function. Though CNN are used in many applications, in this work, we use it for genome classification. Bidirectional long short-term memory (BiLSTM) is an architecture which is built by just combining two independent Recurrent Neural Nets (RNNs) together which makes this network to have forward as well as backward info about the sequence at every instant. GRUs are considered as a variation of the popular LSTM as both are of similar designs and produce equally good results in some cases.

4 Proposed Approach and Architectures

The dataset contains 2524 sequences grouped into 3 classes: MERS, SARS, and SARS-CoV-2, and Table 1 shows the distribution of data. As evident, we have to

Table 1 Number of subsequences under each species

Family name	Number of subsequences in dataset
MERS	804
SARS	1691
SARS-CoV-2	29

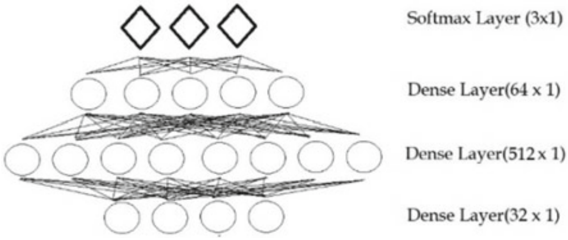


Fig. 1 Multilayer perceptron-proposed architecture

deal with a class imbalance problem and still be able to accurately classify a given subsequence into the correct group.

Since deep learning architectures also deal with numbers, we need to convert each nucleotide into a corresponding vector. Another ideal thing to be done is to find the n-grams of the sequence and vectorize the n-grams using algorithms such as FastText, Word2Vec, and GloVe. However, in the work, the classification model was built by converting each sequence into a number using a dictionary. This dictionary consists of all nucleotide bases present in the sequence and the values in chronological order as present in the universal IUPAC code for nucleotides.

After converting all the sequences into vectors using the value for every corresponding key, the sequences were normalized by padding each sequence with the difference between it and the longest sequence. Subsequently, it was given to the architecture for classification with the corresponding class labels. We propose two types of input characteristics: One based on passing each normalized value of each nucleotide base as a value to each node, and the other one by normalizing the values of k-mers that can be spliced from each sequence. This yields us the single base (SB) and the k-mer models. We have also proposed four architectures based on multilayered perceptron, CNN model, the BiLSTM model, and the GRU model for classification. A pictorial representation for these is shown in Figs. 1, 2, and 3, and the total trainable parameters in each is mentioned in Table 2.

5 Results and Discussion

Though the activation function in each model varies, the overall architecture of a model remains the same in our case. The different activation functions we use in these models are relu, sigmoid, tanh, elu, and selu. There are no activation functions used

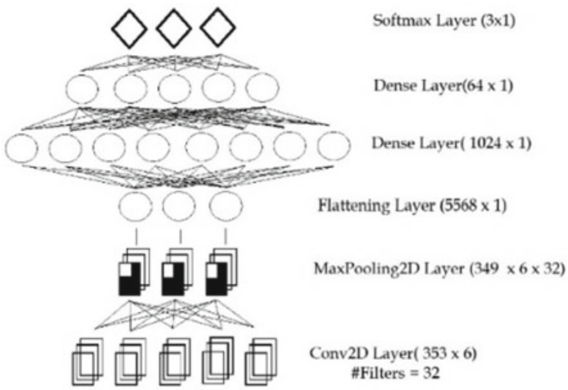


Fig. 2 Convolutional neural network-proposed architecture

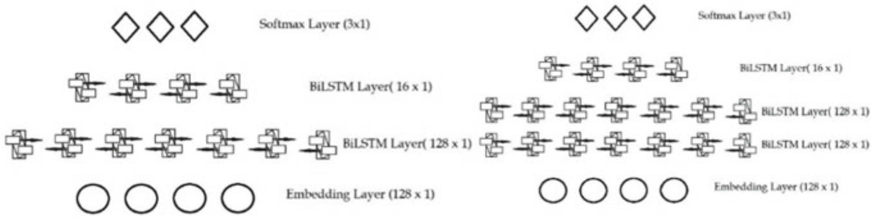


Fig. 3 BiLSTM-proposed architecture (left: 2 layers; right: 3 layers)

Table 2 Total Trainable parameters in each architecture

Architecture	Total trainable parameters
MLP-SB	4,165,635
MLP k-mer	117,731
CNN k-mer	5,769,283
GRU k-mer	532,611
2 Layer BiLSTM	569,315
3 Layer BiLSTM	963,555

for the BiLSTM and GRU architectures. After using these architectures, the results we get can be summarized in Table 3. To find the relation between true-positive and false-positive rate, we can use ROC-AUC curves. For a multiclass ROC-AUC curve, we can consider each class as unique and group all other classes temporarily into another class to evaluate the ROC_AUC curve. Thus, the results we obtain for each architecture after building the model are shown in Fig. 4.

Table 3 Summary of results of all architectures used

Model	Activation	Train accuracy	Test accuracy	Precision	Recall	F1-score
MLP-SB	relu	99.85	94.65	96.67	95.58	96.07
	sigmoid	99.21	93.47	94.95	95.33	95.13
Cross-validated score	tanh	91.53	86.34	89.75	91.76	90.26
=	elu	99.75	94.06	95.31	95.92	95.59
94.44 (±0.79)	selu	99.85	94.06	95.61	95.45	95.53
ROC_AUC_SCORE = 0.9801954						
Model	Activation	Train accuracy	Test accuracy	Precision	Recall	F1-score
MLP-k-mer	relu	96.28	91.96	94.1	92.66	93.3
	sigmoid	85.73	84.27	87.99	87.21	87.57
Cross-validated score	tanh	86.33	85.1	91.24	85.38	87.08
=	Elu	67.4	67.59	22.53	33.33	26.89
91.76 (±0.6)	selu	66.93	67.59	22.53	33.33	26.89
ROC_AUC_SCORE = 0.92975949						
Model	Activation	Train accuracy	Test accuracy	Precision	Recall	F1-score
CNN-k-mer	relu	96.44	92.56	96.11	93.09	94.27
	sigmoid	66.93	66.03	22.01	33.33	26.51
Cross-validated score	tanh	67.63	66.03	22.01	33.33	26.51
=	elu	97.5	92.68	95.87	93.09	94.2
91.76 (±0.6)	selu	97.09	93.88	92.63	94.48	93.33
ROC_AUC_SCORE = 0.950430981						

Model	Train accuracy	Test accuracy	Precision	Recall	F1-score	ROC_AUC score
2 Layer BiLSTM	68.2	66.51	22.17	33.33	26.63	0.5501
3 Layer BiLSTM	66.75	66.51	22.01	33.33	26.63	0.5501
GRU	66.29	68.43	22.81	33.33	27.09	0.4997

6 Appropriate Architecture to Be Used

From these results, we can clearly decipher that the MLP-SB model clearly gives us better results in most cases while the architecture is a bit heavy with several trainable parameters. While the k-mer model was clearly made with the intention of

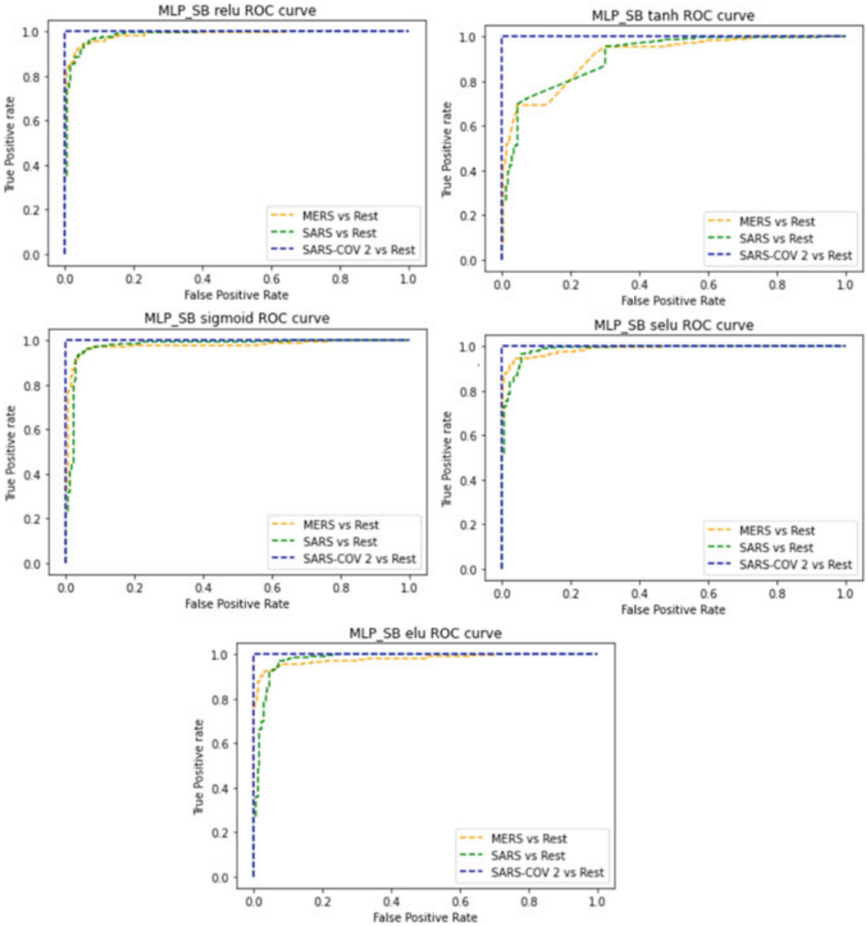


Fig. 4 ROC-AUC curves for MLP-SB architecture {The best results out of all models}

finding a relation between sequences present as a group, the model gives an accuracy slightly lesser than the SB model, but it has significantly lesser training parameters (proportional to the k value we choose to splice the genome). This essentially yields a trade-off: A better accuracy at the cost of more trainable parameters. The CNN architecture has a few 2D convolution layers to relate the k-mers present in the sequence, but present at a distance from the actual k-mer. This adjustment is purely to reduce the model size at the cost of depth of the architecture and find the relationship between non-adjacent k-mers in a sequence. It does ensure that a few activations such as relu, selu, and tanh give respectable results at the end. The BiLSTM architecture does not give very good as expected as the number of layers considered are mush lesser. Intuitively speaking too, the model does not explicitly need to remember certain sequences for classification purposes while training.

However, in future works, the number of hidden layers may be increased, keeping in mind the computational costs, to get good desirable results. So, if the accuracy of BiLSTM can be improved, then GRU's can serve as a good alternative to the BiLSTM architecture with lesser trainable parameters and increased precision.

7 Conclusion and Future Work

Though our approach is quite elegant and simple, there is always a way to improve. Future works can include other approaches which improvise the use of Natural Language processing techniques. In [2] and [3], various combinations of subsequence values of k-mers and size of word embeddings which were non-overlapping nonrepetitive k-mer generation was used to derive good results on MERS and SARS sequences. We can find other systematic approach for sequence representation of MERS, SARS, and SARS-CoV-2 using FastText and Global Vectors (GloVe) [12, 13]-based n-gram embeddings which aims to capture the features of genomes and give significant embeddings. The Word2Vec and Spike2Vec algorithm for creating sequence embeddings are described in [14] and [15]. The suggested ANN-based architecture, called ProtVec, considers a n-dimensional vector representation of the sequence. Authors of [14] and [15] use a near approach to create dna2vec and seq2vec representations, respectively, for sequence embeddings. In order to classify the promoter in a DNA, [16] considered a way of using a mix of continuous FastText n-grams and thus use the embeddings for classification by passing them to a Convolutional Neural Network (CNN) model. This work done can also be extended by using compression techniques on the sequences without losing the structural integrity of the actual sequence. Thus, the process will become more simplified and less utilization of time and resources.

References

1. Saeed U, Usman Z (2019) Biological sequence analysis. In: Husi H (ed) Computational biology [Internet]. Codon Publications, Brisbane (AU)
2. Ganesan S, Kumar SS, Soman KP (2021) Biological sequence embedding based classification for MERS and SARS. Springer Science and Business Media LLC
3. Ganesan S, Kumar SS, Soman KP (2021) Deep learning based NLP embedding approach for biosequence classification. In: MIKE 2021
4. Mikolov T, Corrado G, Chen K, Dean J (2013) Efficient estimation of word representations in vector space. In: Proceedings of the international conference on learning representations (ICLR 2013), pp 1–12
5. Chen W et al (2008) A numerical representation of DNA sequences and its applications
6. Ahmed I, Jeon G (2021) Enabling artificial intelligence for genome sequence analysis of COVID-19 and alike viruses. In: Interdisciplinary sciences: computational life sciences, pp 1–16
7. Randhawa GS et al (2020) Machine learning using intrinsic genomic signatures for rapid classification of novel pathogens: COVID-19 case study. Plos one 15(4):e0232391

8. Gunasekaran H et al (2021) A deep learning CNN model for genome sequence classification. In: Intelligent computing applications for COVID-19. CRC Press, pp 169–185
9. Nguyen N, Tran V, Ngo D, Phan D, Lumbanraja F, Faisal M, Abapihi B, Kubo M, Satou K (2016) DNA sequence classification by convolutional neural network. J Biomed Sci Eng 9:280–286. https://doi.org/10.4236/jbise.2016.95021
10. Rizzo R, Fiannaca A, La Rosa M, Urso A, A deep learning approach to dna sequence classification
11. Mo Z et al (2018) One novel representation of DNA sequence based on the global and local position information. Sci Rep 8(1):1–7
12. George A, Barathi Ganesh HB, Soman KP (2018) Teamcen at semeval-2018 task 1: global vectors representation in emotion detection. In: Proceedings of the 12th international workshop on semantic evaluation
13. George A et al (2019) Significance of global vectors representation in protein sequences analysis. In: Computer aided intervention and diagnostics in clinical and medical images. Springer, Cham, pp 261–269
14. Kimothi et al (2016) Distributed representations for biological sequence analysis. ArXiv abs/1608.05949
15. Ali S, Murray P (2021) Spike2vec: an efficient and scalable embedding approach for covid-19 spike sequences. arXiv preprint arXiv:2109.05019
16. Ng, P (2017) dna2vec: Consistent vector representations of variable-length kmers. arXiv preprint arXiv:1701.06279

The Architectural Design of Smart Embedded Blind Stick by Using IOT

Mayank Gupta, Sweta Jain, and S. K. Saritha

Abstract It is well known that visually disabled people often find it difficult to interact with their nearby environment. In this paper, we propose a design and show the real-time implementation of the smart embedded blind stick. The main components of the Smart Blind Stick comprise Arduino Uno, Ultrasonic Sensor, IR sensor, GPS sensor, and Buzzer. Here, the ultrasonic sensor and IR sensor are used for obstacle detection in the path of blind person, buzzer is used to make the person alert and GPS sensor is used to track the blind person if she or he lost their path. Arduino microcontroller will be used to control the whole scenario, where the smart stick is a closed-loop system to monitor the nearby environment continuously, and send us the output, by comparing the input result in the form of a buzzer sound.

Keywords IOT · Microcontroller · Blind stick · Sensor · Visual impairment · Arduino Uno · Obstacle · Ultrasonic sensor · ETA

1 Introduction

According to the statistical reports, there are 285 million visually impaired people worldwide, of whom 39 million people completely blind, and 246 million having low vision, and there is chance that the number of blind people around the world will rise to double in upcoming years [1]. Visual impairment or blindness refers to the individual losing the function of vision, which occurs when part of the eye or the brain that process images become diseased or damaged. When a person has a vision condition that cannot be corrected by glasses or contact lenses, this is referred to as visual impairment. The Iris of the human eye controls the amount of light that passes through the pupil. It also has a cornea that focuses light and a retina that converts light into nerve signals, that generate an image when delivered to the brain. When the retina or the optic nerve transmits damaged light signals to the brain, vision becomes impossible.

M. Gupta (✉) · S. Jain · S. K. Saritha
Maulana Azad National Institute of Technology, Bhopal, India
e-mail: mayank12451@gmail.com

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_45

While interacting with the real world, the blind person needs the support of either a second person or tools to assist him. For a longer period of time, the conventional navigation systems such as white cans and guide dogs trained by guides or volunteers were used. Day by day, people have begun to use it for building assistive gadgets as technology changes. In addition, such gadgets are built to assist persons with disabilities in their daily lives. And in the recent years, researcher have developed many technologies to take care of the versatility of blind people. The travel aids are typically used to address the mobility of visually impaired persons, with the conventional write-can being the most refreshed and still the most popular. Electronic Travel Aids (ETAs) comprise electronic gadgets, sensors, and signal processors [2–8]. An ETA uses mainly two input components, sensor inputs and camera inputs. Here, sensors are used to take the input in waves or signal form to find out the obstacle, distance of the object, speed of the object, and existence of the object.

With the support of other new technologies such as wireless sensor networks, data analytics, cloud computing, machine learning, and so on, the Internet of Things (IOT) brings revolution by automating the manual process. From medical to education, and agriculture to industry, the Internet of Things is transforming every aspect of life [9]. By collecting and delivering data via wireless sensor networks, sensors serve as the foundation for smart automation. The Internet of Things (IoT) is an excellent technology for solving the concerns that visually impaired persons face. This is what the Smart Embedded Stick proposes to do.

In this paper, the IOT device is being connected with ultrasonic sensor, IR sensor, buzzer, and GPS sensor. The stick analyzes the surrounding area with the help of these sensors, discovers any obstructions in the route, and informs the blind person with a buzzer sound. Apart from this, we have also attached GPS sensor to the blind stick to find out the blind person if she/he gets lost.

2 Components

In the proposed work, following components have been used:

(a) Arduino Uno R3 (ATmega328p)
(b) Ultrasonic Sensor
(c) IR sensor
(d) GPS sensor
(e) Buzzer
(f) battery
(g) wires
(h) Led sensor
(i) Stick

Component description:

(a) **Arduino Uno R3 (ATmega328p)** Arduino Uno R3 is a open-source microcontroller board based on the ATmega328 chip. This microcontroller has 6 analog

I/P pins, 14 digital I/O pins, onboard 16 MHz ceramic resonator, port for USB connection, onboard DC power jack, an ICSP header, and a microcontroller reset button. In order to use the board, simply connect to the computer with a USB cable, or power it with a DC adapter and later by the battery. The programming can be done in the Arduino IDE. The Arduino UNO can be powered either by USB or by an external power supply, and the power source is chosen automatically. Arduino Uno microcontroller is shown in Fig. 1.

(b) **HC-SR04 Ultrasonic Sensor** Sensor can be used to accurately detect objects ranging from very short range (2 cm) to long range (4 m). It can be easily interfaced to any microcontroller. The sensor has two parts, transmitter and receiver. The transmitter sends eight square wave pulses at 40 kHz, and the receiver automatically detects whether the signal is received. A high-level pulse is sent on the echo pin if a signal is returned. The time the signal takes to go from first initiating to the return of the echo is the length of this pulse. As shown in Fig. 2.

VCC is powered by 5 V, while GND is linked to the Arduino's GND pin or to common ground. TRIG and ECHO generate digital signals and are attached to Arduino's digital pins. The distance between the object and the sensor is determined by the time difference between transmitting and receiving the signal. The object's distance from the sensor can be estimated as follows (Fig. 3):

$$d = (v \times t)/2 \tag{1}$$

Here, d = distance, v = velocity, and t = time

(c) **IR Sensor** sensor module adapts to the ambient light. It is made up of two primary parts: a pair of infrared transmitters and receiver tubes. When an IR emitting tube emits a specific frequency, it comes into contact with a reflecting surface. As illustrated in Fig. 5, the IR reflects it back to the receiver tube. The green LED lights up after a comparator circuit is evaluated, and the signal output will output digital signal via the potentiometer knob to modify the monitoring distance . With a working voltage of 3.3–5 V, and the effective range is 2–10 cm (Fig. 4).

(d) **GPS sensor** this proposed work we aim to find out the coordinates of a blind person using GPS so that if a person lost their path so we can find her/him by GPS module. We are integrating an Arduino Uno and a GPS module in this project. The GPS module uses the signals received from the satellite to determine the location (latitude and longitude). Serial communication is used by Arduino to read data from the GPS module (UART) (Figs. 6 and 7).

(e) **Buzzer** sensor and IR sensor periodically provide frequency signals. Signals are reflected back to the source when they contact an obstruction. These signals are received by the sensor's receiver [10]. The sensor sends a signal to the buzzer when items in the range are detected. The magnetic field is attracted by the magnetic flush disk in the buzzer. The oscillation signal generates a changing magnetic field that vibrates at the same frequency as the disk signal when it passes through the coil.

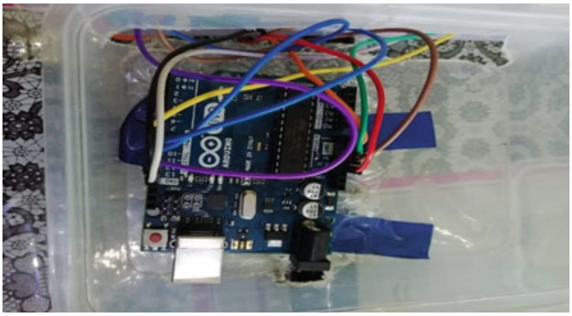


Fig. 1 The Arduino Uno R3 microcontroller board

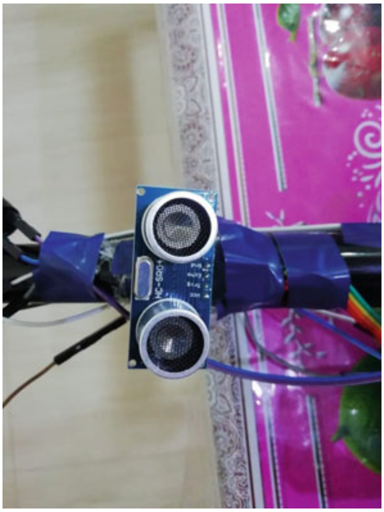


Fig. 2 An ultrasonic sensor

3 Methodology

Following are the key points for the proposed methodology (Fig. 8):

i. Ultrasonic sensor and IR sensor attached to the stick in order to sense the object.
ii. GPS sensor module attached in order to find out the location of blind person.
iii. Calculate the distance of any object from the HCSR-04 and IR sensor using the PIC micro controller.
iv. To find out the location by retrieving the longitude and latitude from GPS module using the PIC micro controller.
v. C programming language.
vi. If the stick finds the obstacle on the way, alerts the blind person by buzzer sound.

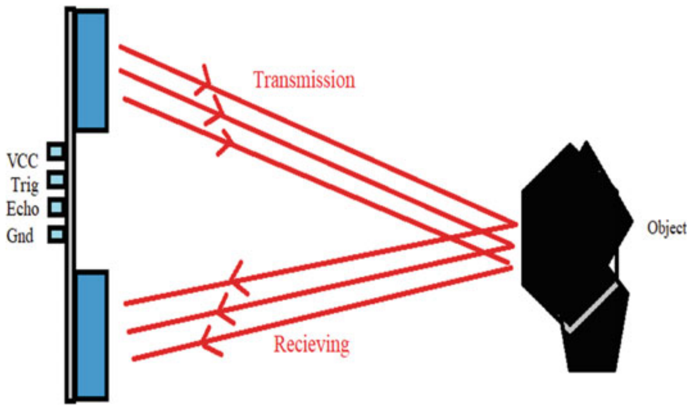


Fig. 3 Working of an ultrasonic sensor

Fig. 4 IR sensor

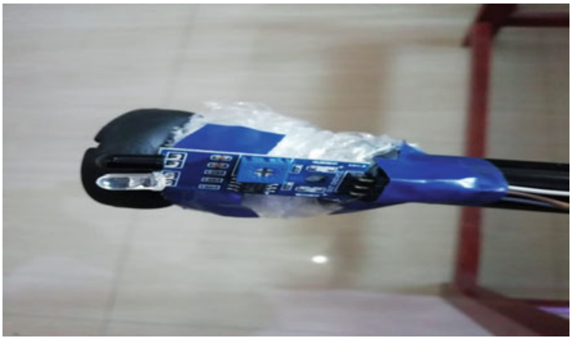
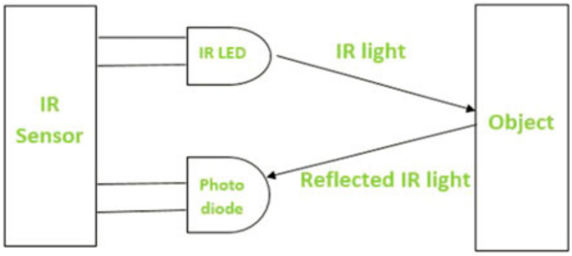


Fig. 5 IR sensor working



A closed-loop system is the smart blind stick (shown in Figs. 9 and 10). The system directs the blind person's movement based on the output of the ultrasonic and infrared sensors, and the difference of which is called as the error signal, on the basis of which the Arduino generates the control commands, which are then fed into the program section, that in turn signals if the buzzer should be sound or not.

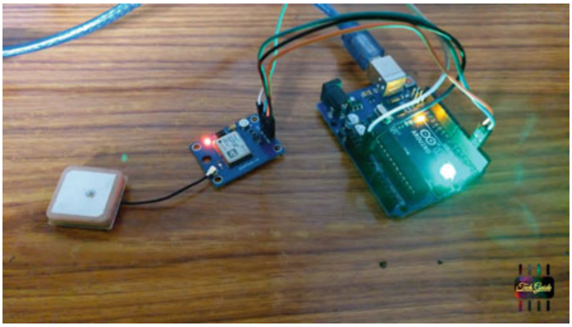


Fig. 6 GPS sensor

Fig. 7 Buzzer

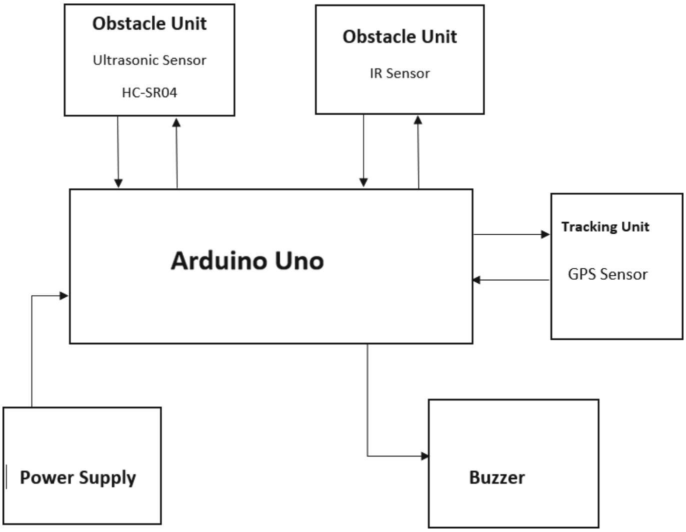


Fig. 8 Block diagram of proposed work

Fig. 9 Blind stick's final prototype

Fig. 10 System from different angle

The buzzer sound and the ultrasonic sensor output are compared, thus deciding the movement of the person. This entire process runs on a loop. For the proposed system, we will be using the Arduino IDE, with additions of special methods and functions. And the smart stick is transformed into an embedded smart stick using this closed-loop control method.

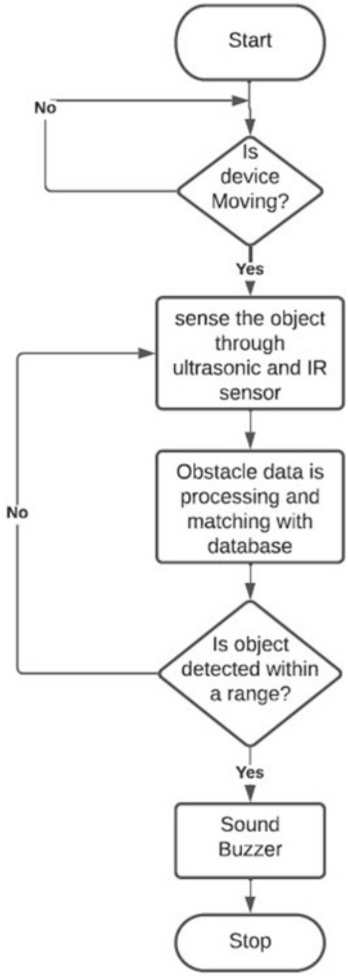


Fig. 11 Flowchart of smart embedded blind stick

3.1 Flow Chart

i. The blind stick is triggered with numerous sensors and a microcontroller when the system is turned on.
ii. When the user moves the blind stick, the sensors connected to it begin sensing the surrounding environment and transmitting the information to the microcontroller.
iii. If case of object detection, it computes the obstacle distance and matches it with database, or else it continues to look for obstacles.
iv. The buzzer is sounded in case of obstacle detection, else continue searching.
v. The similar controlling and sensing power is applicable for all the object detection sensors. The difference is only in buzzer sounds (Fig. 11).

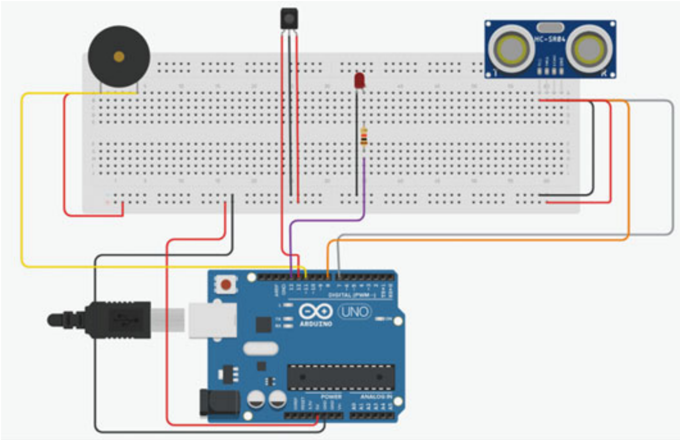


Fig. 12 Circuit Simulation

4 Experimental Setup

We provided a step-by-step approach for connecting all of the hardware components in this part. One Arduino UNO board, an ultrasonic sensor, an IR sensor, a GPS sensor, a buzzer pair, a led, a breadboard, a USB cable, a battery, and connection wires are the components utilized.

The step-by-step process of hardware connections are as follows:

(1) As indicated in circuit Fig. 12, place all of the components on the breadboard.
(2) Construct a common bus A common bus positive BUS +ve and a ground GND BUS.
(3) Connect the Arduino GND to the bus GND and the Arduino 5 V to the bus +ve.
(4) Connect the buzzer pos to Bus +ve and the buzzer neg to Arduino 11.
(5) The ultrasonic sensors' gnd should be connected to Bus GND, and their vcc to Bus +ve. Connect Arduino 7 and Arduino 8 to the echo and trig of the first ultrasonic sensor.
(6) Between the LED anode terminal and Arduino 13, connect the resistor.
(7) Connect LED gnd to Bus gnd anode to resistor side.
(8) Connect the IR sensor's gnd to Bus GND and the IR sensor's vcc to Bus +ve.
(9) Connect the IR sensor output to the Arduino 12.
(10) Connect the USB connection to the computer and use the Arduino IDE to upload the code to the board.
(11) Test the functioning and, if necessary, troubleshoot.
(12) Remove the USB cable and plug in the battery for power.

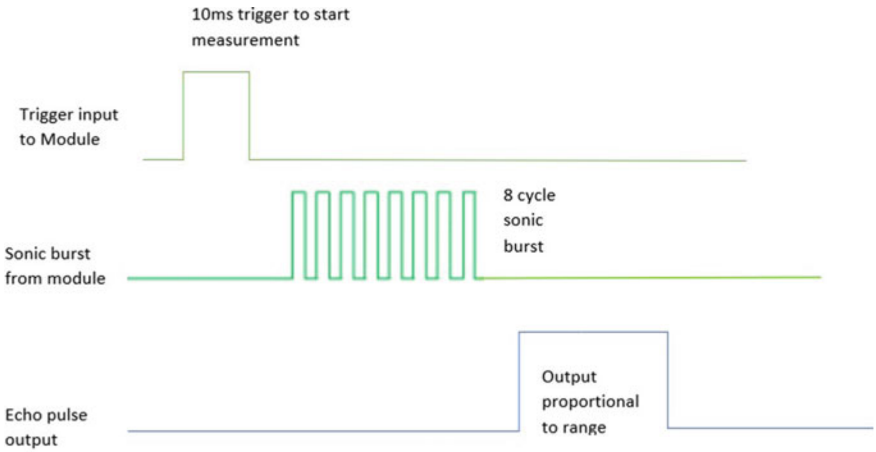


Fig. 13 Timing diagram

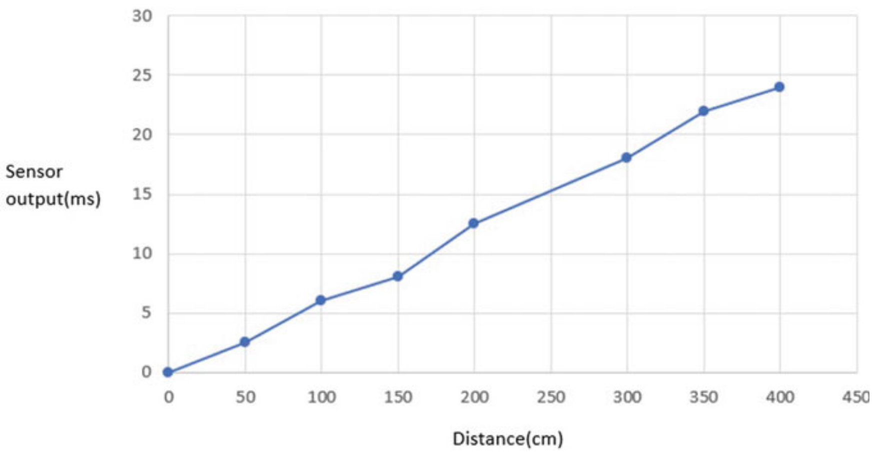


Fig. 14 Time of flight versus measured distance

5 Result and Discussion

Ultrasonic sensors, infrared sensors, PIC microcontrollers, and GPS sensors are all evaluated separately and together. Because ultrasonic and infrared sensors function on the concept of echo, evaluating their reflection on various obstacles is crucial (Figs. 13 and 14).

A pulse of ultrasonic sensor is about to sent 10 Ã^Â¼s to trigger the module. After which, the module automatically sends 8 cycles of 40 KHz. The module is about to be triggered by a 10 Ã^Â¼s pulse from an ultrasonic sensor. After that, the module delivers eight cycles of 40 KHz ultrasound signal and detects the echo. After

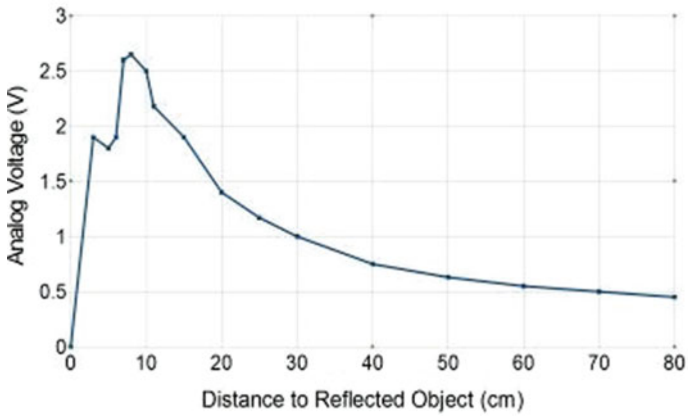


Fig. 15 Analog voltage versus distance of IR sensor

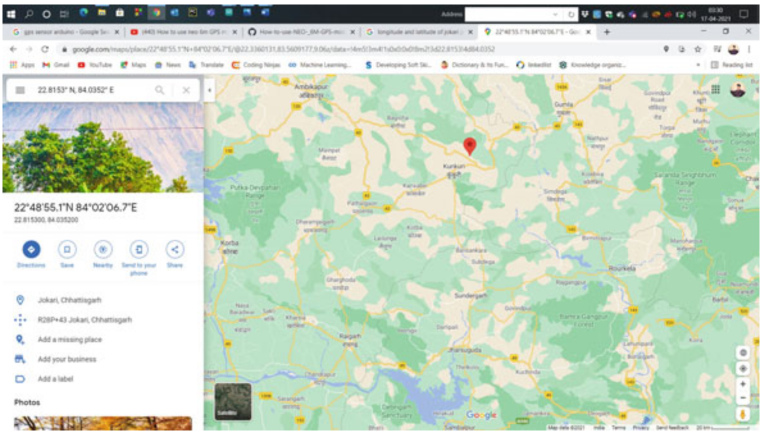


Fig. 16 Position of blind person

colliding with a barrier, the signal bounces back and is caught by the receiver. As a result, the obstacle's distance from the sensor may be easily determined.

Time of flight (TOF) is used by ultrasonic distance sensors to identify obstacles; the output is a digital pulse with a duration equal to the time it takes for the sound to reach the target and return. Several tests were conducted on various obstacles at various distances, and the graph below shows the relationship between distance and sensor output (the average of TOF).

We put the ultrasonic sensor to the test in real time and compared the results to simulated calculations. Table 1 shows a comparison of the ultrasonic sensor analog signal value between the observed and calculated values, allowing us to spot errors. Same as we have performed test on IR sensor at different distance to observe the

Table 1 Result of ultrasonic sensor comparison

Range (cm)	Calculated analog value (mV)	Measured analog value (mV)	Error
0	0	0	0
5	25	24.1	0.9 mV
20	100	98.6	1.4 mV
40	200	194.3	5.7 mV
50	250	244.15	5.85 mV

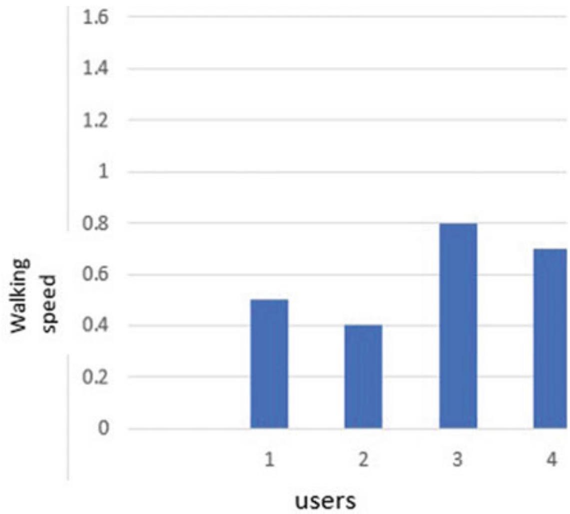


Fig. 17 Performance and accuracy of blind stick

analysis in between IR sensor voltage and distance to reflected object, which is shown Fig. 15.

In case of any emergency, if user lost their path, we have implemented GPS sensor with Wi-Fi module Node MCU esp32 and tested well. To handle this, there is an Android application called Blynk to retrieve the position of blind person as shown in Fig. 16. To test the performance of the stick in the real world, two trained and two untrained blind people, with a few obstacles, were used, where two of them are aware with the blind stick and two are not.

A blind person was asked to walk through various obstacles in the testing area. The individual's walking speed is monitored. As shown in Fig. 17, the time it takes for trained and untrained people to successfully walk through the obstacles is calculated. The average speed of trained users was 0.75 m/s, while that of untrained users was 0.45 m/s, according to our calculations. In comparison, sighted people's travel speed is (1.4 m/s). The results show that using the blind stick to train the user improves his walking speed as well as his confidence in avoiding obstacles.

6 Conclusion

Proposed system is novel approach for visually impaired people for providing navigation in both indoor and outdoor environment. This prototype is simple to use, inexpensive, comprehensible, and very cost effective. Thus, every people can effort it. When the smart blind stick detects an obstacles in the route of the concerned person, it sounds a buzzer to notify them. In addition, the installed system identifies any obstruction within a 2-m range. And throughout our study, we have focused on one issue, vision impairment. We performed this low-cost project to come up with a solution. We hope that this project will expand across society, transforming disabled people into capable people. This is our hope, to use this stick as a smart eye for those with vision problems [11, 12].

References

1. World Health Organization et al (2019) World report on vision
2. Zeng L, Simros M, Weber G (2017) Camera-based mobile electronic travel aids support for cognitive mapping of unknown spaces. In: Proceedings of the 19th international conference on human-computer interaction with mobile devices and services, pp 1–10
3. Ranaweera P, Madhuranga S, Fonseka H, Karunathilaka D (2017) Electronic travel aid system for visually impaired people. In: 2017 5th international conference on information and communication technology (ICoIC7). IEEE, pp 1–6
4. Obermoser S, Klammer D, Sigmund G, Sianov A, Kim Y (2018) A pin display delivering distance information in electronic travel aids. In: 2018 7th IEEE international conference on biomedical robotics and biomechatronics (BIOROB). IEEE, pp 236–241
5. Laubhan K, Trent M, Root B, Abdelgawad A, Yelamarthi K (2016) A wearable portable electronic travel aid for blind. In: 2016 international conference on electrical, electronics, and optimization techniques (ICEEOT). IEEE, pp 1999–2003
6. Ferrand S, Alouges F, Aussal M (2020) An electronic travel aid device to help blind people playing sport. IEEE Instrument Measur Mag 23(4):14–21
7. Bujacz M, Strumiłło P (2016) Sonification: review of auditory display solutions in electronic travel aids for the blind. Arch Acoust 41(3):401–414
8. Erp JBV, Paul KI, Mioch T (2020) Tactile working memory capacity of users who are blind in an electronic travel aid application with a vibration belt. ACM Trans Accessible Comput (TACCESS) 13(2):1–14
9. Kumar S, Tiwari P, Zymbler M (2019) Internet of things is a revolutionary approach for future technology enhancement: a review. J Big Data 6(1):1–21
10. Parihar V, Rohilla Y, Kumari K (2020) Ultrasonic sensor based smart cap as electronic travel aid for blind people. In: 2020 third international conference on smart systems and inventive technology (ICSSIT). IEEE, pp 873–877
11. Robu IR sensor working. https://robu.in/ir-sensor-working/
12. Elprocus, buzzer sensor working. https://www.elprocus.com/buzzer-working-applications/

Optimizing CNN Architecture Using Genetic Algorithm for Classification of Traffic Signs in Real Time

Ruchika Malhotra, Saanidhi, and Dev Gupta

Abstract With the notion of smart cities transforming cities into digital societies and making people's lives easier in every way, Intelligent Transportation Systems have become an integral element among all. The Intelligent Transportation System (ITS) attempts to improve traffic efficiency by reducing congestion and ensuring the safety and comfort of commuters in real time. Traffic sign detection and recognition is one of the multifaceted conjunctive fields of research in ITS. In this paper, we address the issue of the TSR (traffic sign recognition) problem, i.e., classification of traffic signs along the roadside which plays a crucial role in developing advanced driver assistance and autonomous driving systems. CNN's network design has a huge impact on its performance and convergence. As a result, we use the Genetic Algorithm (GA) to automate the task of selecting a high-performance CNN (Convolutional Neural Network) Architecture for the GTSRB (German Traffic Sign Recognition Benchmark) dataset. The model is optimized through GA using multiple network configurations in the search space. Our model takes into account the limitations of the dataset, and we use certain data augmentation approaches to address the issues. We were able to attain an average accuracy of 98.2% which demonstrates the state-of-the-art performance on the publicly available dataset.

Keywords Convolutional neural network · Genetic algorithm · Multiclass classification · Deep learning · Model optimization

R. Malhotra · Saanidhi (✉) · D. Gupta
Delhi Technological University, New Delhi, India
e-mail: saanidhi.1211@gmail.com

R. Malhotra
e-mail: ruchikamalhotra@dtu.ac.in

D. Gupta
e-mail: devgupta_2k18it042@dtu.ac.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_46

1 Introduction

Advanced driver assistance systems are designed to enhance car components to improve safety and driving performance. Road sensors, in-vehicle navigation services, electronic message signs, traffic management, and monitoring, and so on are examples of these systems. One such feature is traffic sign detection and recognition to make specific judgements to avert the potential risks. The feature must be robust to a variety of real-life situations such as low illumination, obstructions, or the sign being far away.

With the advancement of image processing techniques, a variety of deep learning-based technologies have recently been applied to a range of recognition tasks. The major objective is to devise a way for picking the optimal configuration for each layer of a deep learning model. GA is useful for issues that require sophisticated algorithms to solve since it seeks the optimal solution by simulating the natural evolution process. When compared to state-of-the-art classification networks, the algorithm-configured network structure performed similarly. In this paper, we show that our version of CNN determined by GA from the exhaustive search space consisting of network configurations performs exceptionally well on the given input dataset.

The article is organized as follows: Section 2 summarizes the related work that has already been done on this subject. Section 3 contains the specifics of the model structure suggested in this study, as well as other pertinent information. The evaluation metrics, experimental findings, and task discussions are presented in Sect. 4. Section 5 brings the paper to a conclusion.

2 Literature Review

On the themes of TSR and TSD (Traffic Sign Detection), there is a massive amount of literature, and numerous research papers available. These papers use ConvNet-based approaches to solve the task.

The paper [1] uses a Deep Neural Network which comprises Convolutional layers and Spatial Transformer Networks. It performs a comprehensive analysis of different adaptive and non-adaptive stochastic gradient descent optimization algorithms such as SGD, SGD Nesterov, RMSprop, and Adam, along with multiple combinations of Spatial Transformers on the GTSRB dataset and achieves the top-ranked accuracy of 99.71%. Another article [2] suggested a unique technique of Separating-Illumination Network (Sill-Net)-based CNN. This method comprises three major steps starting with the segmentation of the illumination features and semantic features of the image using the disentanglement method. A repository is created by combining the lighting features. The repository is then imported into the support samples, resulting in an improved training set that is subsequently utilized to train a recognition model. This model was then evaluated on a variety of classification task datasets including GTSRB and TT100K, generalized few shot benchmarks miniImageNet, CUB, CIFAR-FS, and three other logo datasets. In most situations, it outperformed other state-of-the-art techniques by a significant margin. The paper [3] used the GTSRB dataset for training and the Indian dataset for testing to detect the images of Indian Traffic signs on the road using CNN. For traffic sign detection, their approach incorporated transfer learning from trained models—YOLO v3-v4 and BLOB detection. For classification, they used the most efficient filter after testing the CNN models with filters of different dimensions. They achieved a high training accuracy of 98.850% and testing accuracy of 86.880%. The authors offer the Genetic CNN technique in [4] to automatically design architecture to effectively deal with the image classification purposes. The proposed algorithm was validated on CIFAR10 and CIFAR100 benchmark datasets where CIFAR100 was used to show superiority of their algorithm as others do not perform their experiments on it due to its large number of classes. The proposed algorithm outperforms the accuracy of existing automatic CNN by giving 96.78% on CIFAR10 and 79.47% on CIFAR100.

3 Proposed Work

3.1 Data Exploration and Description

The GTSRB [5] used in this paper is one of the widespread datasets randomly obtained from the camera in a real-time scenario and was first presented at the International Joint Conference on Neural Networks (IJCNN) in 2011. The dataset consists of 51,839 images of German traffic signs divided into 43 categories. Figure 1 represents the data distribution—the training set comprises 34,799 images (67.1%), validation set has 12,630 images (24.4%) and test set includes 4410 images (8.5%).

3.2 Data Preprocessing

The dataset poses 2 major difficulties in classification—low resolution with poor contrast and uneven class distribution. The low contrast images could be identified using the inbuilt functionality provided by OpenCV [6]. According to the *is_low_contrast* function, an image is a low contrast image if the range of brightness in the image's histogram plot is concentrated only over a certain region and spans less than a fraction of the full range.

The class label skewness hampers the overall classification accuracy, labeling accurately for the category which contains more data and identifying inaccurately for the underrepresented classes. The abovementioned issues in the dataset could be resolved by applying various techniques elaborated in this section.

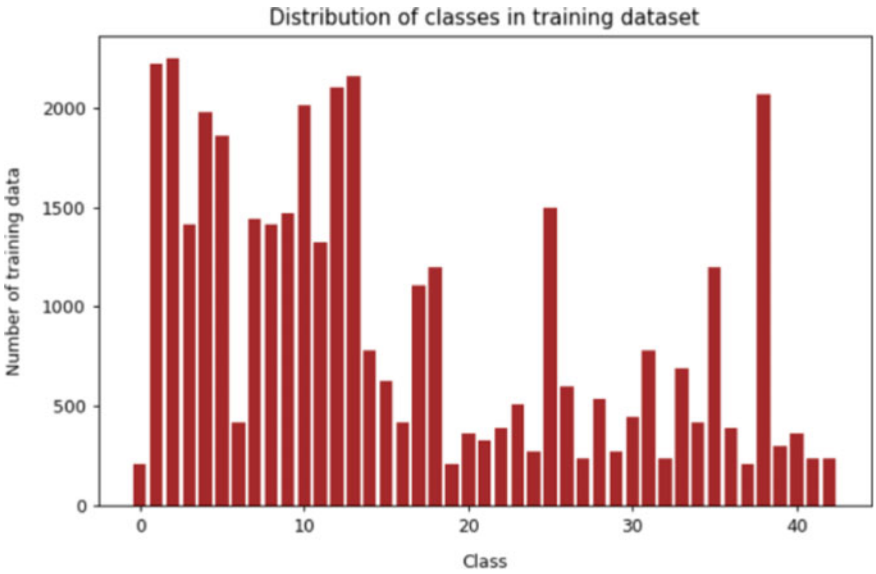


Fig. 1 Depicts the distribution of images across each unique class and suggest that the data is divided in extremely irregular fashion

Data Augmentation

Data augmentation is a technique used for image expansion to improve the model generalization ability. In this method, training data is converted into small batches which are subjected to a random selection algorithm used for choosing a data augmentation technique from a pool. This step is performed to enhance the learning capability of the network and improve the accuracy of identifying the target object. It introduces a low information distortion level; therefore, the class labels will remain the same. The image preprocessing is done using the OpenCV, a real-time optimized Computer vision library tool and hardware (Fig. 2).

Local Histogram Equalization—This method is used to enhance an image with low contrast by spreading out the most frequent intensity values in an image.

Random Cropping and Filling—It is a technique wherein we create a modified subset of an original image. The desired output area of the image is set, then the random output coordinates are determined, and the crop is performed.

Bilateral Filtering—A bilateral filter is for reducing noise, smoothing out the image, without distorting the edges. It displaces the intensity of each pixel with a weighted mean of intensity values from the neighboring pixels.

Color Transformation—It is concerned with processing the components of a color image within the framework of a single-color model.

Rotation—This technique is performed to rotate the training images by a minute angle, 10°, in our case to improve the classification ability of the network.

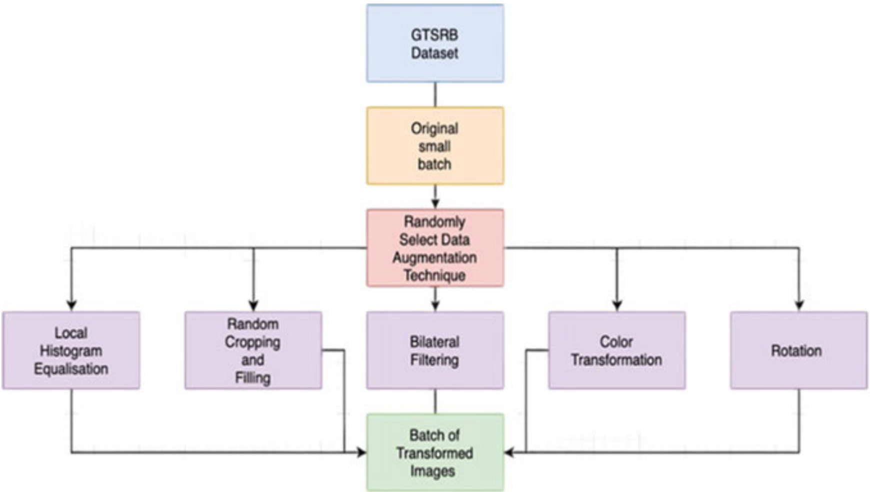


Fig. 2 Data augmentation

Resolving unequal class distribution

To fix the unequal class distribution challenge, the number of instances per category is made to be equal to a random number which is determined by examining the dataset. According to Fig. 1, Class 2 has a maximum number of instances, i.e., 2010 instances. Hence, the random number chosen can be 4000 which is approximately double of 2010.

3.3 *Hyperparameters*

Optimization Algorithm and Activation function comprise the core components of the network architecture of CNN. The hyperparameters directly control the behavior of the algorithm, increment the convergence speed of the network, and enhance the performance of the model by averting overfitting. The activation function is a non-linear modification that is applied to the input before transferring it to the next layer of neurons or converting it to output. Rectifier Linear Unit (ReLU) [7] is one of the most widely used activation functions. It is defined as:

$$R(z) = \max(0, z) \tag{1}$$

Another activation function is Softmax which is often regarded as a combination of multiple sigmoids. It is represented by:

$$\sigma(\vec{z})i = \frac{e^{zi}}{\sum_{j=1}^{K} e^{z_j}} \quad (2)$$

It is used for multiclass classification [8] problems to determine the probability of the test data belonging to each class at the end of the network. An optimization algorithm is a process that compares several solutions iteratively until an optimum or suitable one is discovered.

3.4 Model Architecture

Outstanding Network Architecture is necessary for a deep learning model to outperform other state-of-the-art DNNs. The Network Architecture Search process is automated using Machine Learning techniques. Genetic CNN is one such method that selects CNN configurations using the GA. The link between the convolutional layers of the initial pool of candidates is encoded in this method. Furthermore, the algorithm selects CNN candidates with the network's best fitness score performance. Through crossover, the chosen ones produce offspring, and mutation is then applied to a random selection of candidates. To find an effective network architecture structure for a given input, we employ network configurations as the search space in this paper, keeping the other hyperparameters constant due to high computational overhead costs (Fig. 3).

The connections between convolutional layers are encoded using a binary representation method. The network structures are divided into stages, with convolution layers between the input and the pooling layer at each stage. The layers within a

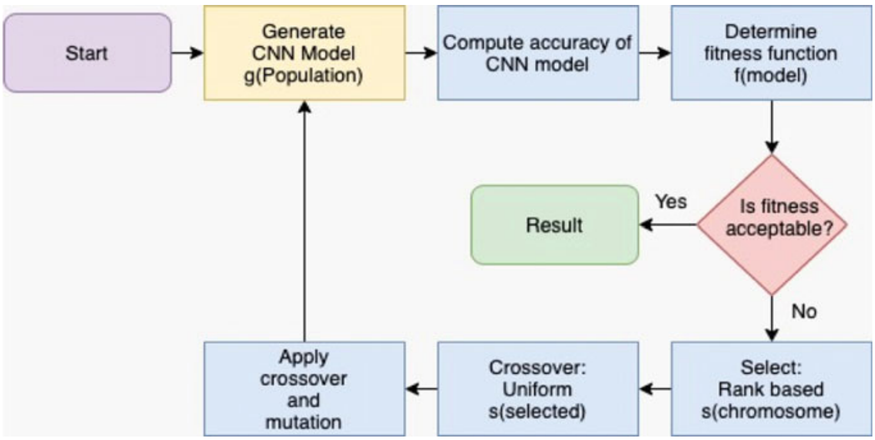


Fig. 3 Block diagram depicting the process of our algorithm

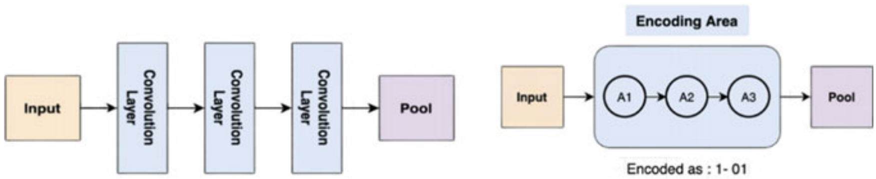


Fig. 4 Illustrates the structure with one stage having three nodes within the stage ($Ks = 3$). The number of bits required to encode the structure = 3

stage are termed nodes. To ensure that fixed-length binary strings are encoded, this approach is used to establish a family of networks.

A network is composed of S stages, $s = 1, 2, \ldots, S$, and the sth stage contains K_s nodes, denoted by v_s, k_s, $k_s = 1, 2, \ldots, K_s$. The nodes within each stage are ordered, and only forward connections are permitted. The connections are converted into binary numbers, encoding the summation of preceding input nodes for each node. A bar is used to separate the stages, while a hyphen is used to divide the nodes in the binary string. The number of bits required to encode the inter-node connections of a network with S stages, where each stage has ks nodes is determined by the formula (Fig. 4):

$$\text{number of bits} = \sum_{S} \frac{1}{2} K_s (K_s - 1) \tag{3}$$

3.5 *Process*

The initialization step generates a population of CNN models in encoded, binary format by randomly creating a set of candidates. Each candidate in the population is referred to as a chromosome, abbreviated as s. The next step evaluates the accuracy of categorization of the decoded chromosome with each of the supplied populations of network architectures. These accuracies are then utilized to determine the algorithm's fitness. The fitness score is an objective function responsible for determining the survival of each chromosome. For our algorithm, an optimal fitness score is chosen, and the process is repeated until a CNN model reaches the score. If no network configuration achieves the ideal fitness score, the algorithm follows the below-mentioned steps as shown in Fig. 3. In Rank-based selection system, the chromosomes with better classification accuracies will survive this stage to guarantee that the algorithm converges quickly. During Uniform Crossover, the individuals who survived the previous stage develop the next generation of individuals. Using a random variable, a random pair of parents is chosen to produce offspring. The mutation process for an individual chromosome consists of flipping each bit independently with a certain probability to sustain the good properties of surviving individuals.

Table 1 Results of genetic CNN, showing accuracy of the generated networks

Number of generations	Maximum accuracy %	Minimum accuracy %	Average	Best network architecture
50	98.8	97.6	98.2	1–01 | 1–11-011–0111 | 1–11

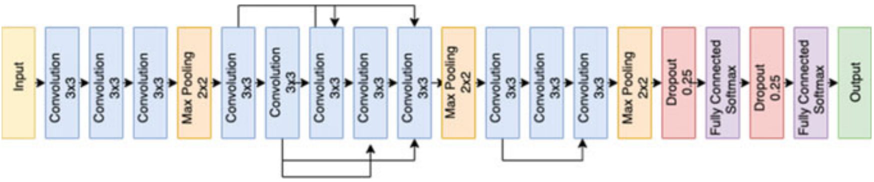


Fig. 5 Depicts the final network architecture

4 Results and Discussion

In our paper, we formulated a neural network with $S = 3$ and nodes $(K1, K2, K3) = 3, 4, 3$. The total number of bits required to encode the network could be calculated using the equation. The total length of chromosome = 12.

Other parameters like the number of individuals N and number of generations were set to 10 and 30, respectively. To determine the network's accuracy, the selected chromosome was decoded into the network and trained for 50 epochs. The fitness value for the selection process was assumed to be equal to the accuracy value. The evaluation metric chosen to assess model performance was accuracy (Table 1; Fig. 5).

It comprises 11 convolutional layers divided into 3 stages and 3 max-pooling layers. A dropout layer is introduced at the end of the pooling layer and after a fully connected layer. We have used the softmax activation function for the output layer since it is a multiclass classification problem. We used GPU for faster processing of the model.

The proposed CNN model architecture which is chosen as a result of selection through GA after 50 generations were able to achieve high accuracy of 98.2% which is comparable to other state-of-the-art architectures.

5 Future Work and Conclusion

In this paper, we developed a CNN architecture using the optimization capability of the GA for a given dataset. We attained an average accuracy of 98.2% with a network consisting of only 12 weighted layers. In the future, we hope to modify our algorithm to operate with a variety of input datasets. We can also incorporate hyperparameters like activation function and optimization algorithm in the search space for genetic algorithms to further enhance the model's performance. Some other preprocessing techniques could also be used to improve on the raw dataset end. We can also try comparing performances with other models like ResNet [9] and VGGNet [10].

References

1. Arcos-García Á, Alvarez-Garcia JA, Soria-Morillo LM (2018) Deep neural network for traffic sign recognition systems: an analysis of spatial transformers and stochastic optimisation methods. Neural Netw 99:158–65
2. Zhang H, Cao Z, Yan Z, Zhang C (2021) Sill-net: feature augmentation with separated illumination representation
3. Bichkar M, Bobhate S, Sonal C (2021) Traffic sign classification and detection of indian traffic signs using deep learning. Int J Sci Res Comput Sci Eng Inf Technol, pp 215–219. https://doi.org/10.32628/CSEIT217325
4. Sun Y, Xue B, Zhang M, Yen G, Lv J (2020) Automatically designing CNN architectures using the genetic algorithm for image classification. IEEE Trans Cybern, pp 1–15. https://doi.org/10.1109/TCYB.2020.2983860
5. Houben S, Stallkamp J, Salmen J, Schlipsing M, Igel C (2013) Detection of traffic signs in real-world images: the German traffic sign detection benchmark. In: Proceedings of the international joint conference on neural networks. https://doi.org/10.1109/IJCNN.2013.6706807
6. Aghdam H, Heravi E, Puig D (2017) A practical and highly optimized convolutional neural network for classifying traffic signs in real-time. Int J Comput Vision 122:246–269. https://doi.org/10.1007/s11263-016-0955-9
7. Nair V, Hinton G (2010) Rectified linear units improve restricted Boltzmann machines. Proc ICML 27:807–814
8. Stallkamp J, Schlipsing M, Salmen J, Igel C (2011) The German traffic sign recognition benchmark: a multi-class classification competition. In: Proceedings of the international joint conference on neural networks, pp 1453–1460. https://doi.org/10.1109/IJCNN.2011.6033395
9. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition, pp 770–778. https://doi.org/10.1109/CVPR.2016.90
10. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv 1409.1556

Analysis of Student Satisfaction on Virtual Learning Platforms During COVID-19

K. Abirami and G. Radhika

Abstract As a result of the global spread of COVID-19, e-Learning has recently experienced extraordinary growth. Many educational sectors have made the transition from traditional classroom learning to virtual learning via various online platforms. In this epidemic, virtual learning has enabled all schools and universities to continue to provide education. This rapidly growing alternative modality necessitates the provision of robust and high-quality education. It is also important to figure out whether online learning satisfies the needs of pupils. Even if learning has become easier, many people still confront difficulties, poor connectivity and e-platform. This study aims to identify the students' satisfaction by conducting a survey and analyzing it by means of data analysis and data visualization.

Keywords Google cloud platform · Sentiment analysis · Visualization · WordCloud

1 Introduction

Most educational institutions throughout the world have been shuttered since about March 2020 to slow the spread of the COVID-19 epidemic. The lockdown due to the pandemic in India has brought an interruption in the education of students. The offline mode of education has been flipped, and the virtual mode of learning is the new trend. Online learning provides an efficient way of teaching. The potential benefits of implementing a Virtual Learning Environment have been recognized by educational institutions. This is because the Internet creates, fosters, delivers, and facilitates learning, anytime and anywhere. E-learning is also made easier and more interesting. It also gives more individualized, comprehensive, dynamic learning content

K. Abirami (✉) · G. Radhika
Department of Computer Science, Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, India
e-mail: k_abirami@cb.amrita.edu

G. Radhika
e-mail: g_radhika@cb.amrita.edu

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_47

in real-time, aiding in the development of knowledge. The coronavirus (COVID-19) pandemic threw this situation into disarray and forced e-learning to take center stage. Face-to-face lectures were replaced with either online lectures or other materials. Students began to communicate via digital channels to complete group tasks and even socialize [1]. To achieve this objective, student satisfaction, instructor support, student interaction and collaboration, and course materials were key factors to be assessed. The online mode of education, being new to everyone, has its own merits and demerits. This research will gather feedback from students in high schools and colleges about their virtual learning experiences when, because of educational institution closures, an unprecedented quantity of learning began to take place online. The fundamental research topic of this study is to examine e-learners' perceptions of quality issues in the entire e-learning process. Further, data analysis will be made to assess the satisfaction of students in online educational platforms. Finally, some innovative solutions and recommendations are provided to make virtual learning student friendly. We propose to analyze the Student Satisfaction with Online Education Platforms in India during the COVID-19 pandemic. It has caused a sudden shift in the teaching and learning process in schools and colleges. When schools and colleges all around the world had to be shut due the outbreak of COVID-19 pandemic, online learning became an urgent necessity to ensure that the students had a continuity in their education. Though online learning serves as a great tool in personalized and visual learning experience, there are quite a lot of challenges associated with it. Our study focuses on analyzing various concerns of students in the virtual learning process and aims to provide a solution.

2 Related Works

Analysis of User Satisfaction with Online Education Platforms in China during the COVID-19 Pandemic has been presented in [2]. This paper talks about a survey conducted on impacts of virtual learning and how they have extracted the factors influencing satisfaction and established a scientific and effective satisfaction index system. Also, the data obtained from an offline questionnaire were examined and analyzed, and a structural equation model was built for quantitative analysis of the relationship between various indicators.

E-Learning: Survey on Students' Opinions [3]. This paper thus concentrates on the survey from which reliable data is obtained regarding the student's opinion of e-learning, what their real predispositions for e-learning are, what form of course materials they expect, which type of questions they find most useful for knowledge assessment, the way course materials are presented, types of questions for knowledge assessment, etc. Then, an analysis is carried out and results are presented and analyzed.

A survey of sentiment analysis in social media. Sentiment Analysis of Twitter Data: A Survey of Techniques [4]. The users post their tweets in Twitter. These tweets are extracted in the form of unstructured data. The unstructured dataset is converted into structured form and then extracts features from structured review. The features of the words are selected and then classification technique is applied on extracted features to classify them into its sentiment polarity that is namely either positive or negative [5, 6]. Online social networks are to share content with maximum users. This enormous sharing of information can lead to malicious users leverage the privacy of an individual.

An attempt has been made in this paper to improve privacy preservation by calculating the reliability of a user and can entrust on his/her peer using the content sharing, machine learning, and tone analysis [7]. This paper elaborates a Smart Cloud E-Learning System with Social Networking based on architecture level of social networking and e-learning cloud system to support e-learning interactions in world-wide environments [8, 9]. Here, a survey is conducted and a comparative analysis of existing techniques for opinion mining like machine learning and lexicon-based approaches is done. Also, general challenges and applications of Sentiment Analysis on Twitter is discussed. In our proposed work, we have focused on analyzing the concerns of students and providing a solution for them. The fundamental research topic of this study is to examine e-learners' perceptions of quality issues in the entire e-learning process. We also made various data analyses, Sentiment analysis visualizations using different Python libraries. We have also produced a simple solution of chrome extension to disable unwanted access. Twitter tweets regarding e-learning during COVID are extracted using Tweepy and analysis will be done using Python's TextBlob library [10]. In this paper, they have taken different datasets from Facebook and twitter and have analyzed it with various sentiment analysis techniques based on modularity.

Jupyter Notebook—For performing various analysis and visualizations, Python libraries—To analyze and visualize the data.

Google Cloud Platform—For performing analysis on the data.

3 System Requirement Analysis

- Data Collection
- Static Data Analysis and Visualization
- Sentiment Analysis in Python and Google Cloud Platform
- Sentiment Analysis on Twitter Data in Python (Streaming Data)
- Recommendations and Solution Implementation.

3.1 Proposed Methodology

Module details of the system.

We split the module into six sub-parts as mentioned in Fig. 1.

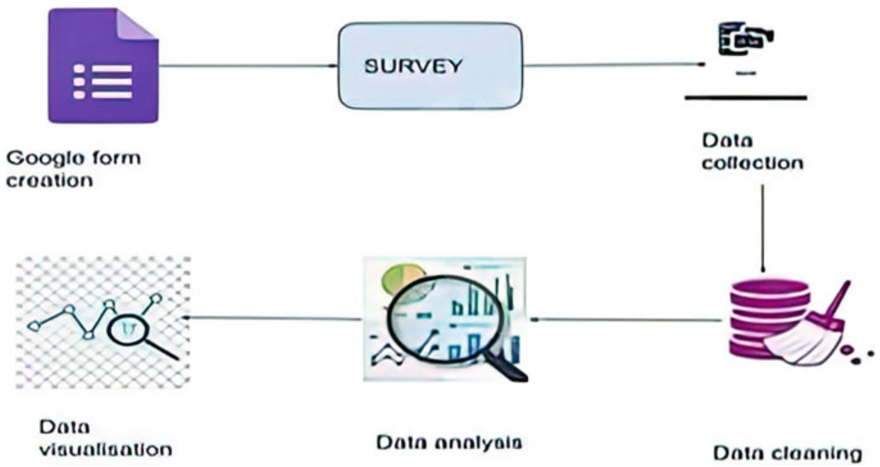


Fig. 1 Architecture diagram

- Data Collection: Data collection from the survey conducted via Google forms a Twitter Static Data
- Analysis and Visualization: Exploratory Data Analysis and Data Visualization using different Python libraries like NumPy, Pandas, Matplotlib, etc.
- Sentiment Analysis in Python and Google Cloud Platform (Static Data): Here, we have made use of Natural Language Processing libraries (TextBlob, WordCloud) in Python framework Jupyter Notebook to do a detailed sentiment analysis on the overall experience of users in online learning.
- Sentiment Analysis on Twitter Data in Python (Streaming Data): Here, a sentiment analysis is made on all the tweets regarding online learning during this pandemic using a NLP library called Tweepy.
- Recommendations and Solution Implementation: A simple solution of a chrome extension is designed to disable access to all other unwanted web sites during online classes.

First, a list of questionnaires is formulated and then a google form is created. Then, a survey is conducted via Google Forms to collect the responses from students. The survey data collected is then cleaned to remove noisy data. Then, various types of analysis are done on it to get some valuable insight from the survey. Then, data is visualized in the form of bar charts and various plots. Then depending upon the analysis carried out, various recommendations are given and some solutions are provided to improve student experience in online learning. Then, streaming tweets from Twitter about e-learning in COVID-19 is collected and then analyzed.

4 Implementation and Testing

4.1 Data Collection

The questions are populated regarding the pros and cons of e-learning, and a survey is conducted using Google Forms. After the survey is taken by parents and students, the required data is collected. The form data is then exported to .csv format for analysis.

The link for the Google Form survey conducted is:
docs.google.com/forms/d/1IK9UTzmzVBpmpEmsjJrRfNxaJSbO6HSpIXgz7jg UJI/edit?usp=sharing.

The link for the dataset is:
docs.google.com/spreadsheets/d/1HbS1LPVWaVBuVYz3u7JynLUSxfXYo RiTsUxcOPFxbgY/edit?usp=sharing.

4.2 Data Cleaning and Feature Extraction

This type of analysis is done by using Natural Language Processing (NLP) API in Google Cloud Platform. This API reveals the structure and meaning of text which is mainly used to extract information about people, places events in text documents, blogs, etc., analyze the sentiment of the text, and parse intent from customer conversions in messages followed by (1) data encoding is done to convert the categorical values to numerical for easier analysis using techniques like one hot encoding and label encoding. (2) We have visualized the data to get some insights regarding it using various python visualization libraries like seaborn, Matplotlib. Visualization is done using various types of plots like histogram, pie charts, and boxplots. Sentiment Analysis is done starting with Tokenization: When analyzed textual data, tokenization is a very crucial step. By evaluating the sequence of words, tokenization aids in interpreting the meaning of the text. Basically, tokenization can either be word tokenization or sentence tokenization. In our application, we have used word tokenization (breaking of a sentence into words). Tokenization can be done using a variety of methods and libraries. Some of the libraries that can be utilized to do the work include NLTK, Gensim, and Keras. Tokenization is handled using NLTK (The Natural Language Toolkit) in our application. Here, NLKT is chosen because it provides the fastest and most accurate syntactic analysis over any other Natural Language Processing library released till date. Stop words (words in the text that add no meaning to the sentence and whose removal will not impact the p) should be removed after tokenization. Stemming: In our sentiment analysis application, we want all the tenses of a word to be treated the same because the sentiment of the word does not differ in different tenses of the same word. We want the program to recognize that the words "ask" and "asked" are just different tenses of the same verb. Words like "learn," "learned," and "learning" should be treated as the same. In short, all tenses of the word should be reduced to its root word. This is achieved by using a technique called stemming [11]. The idea here is to reduce different forms of a word to a core root. Words that are derived from one another, especially if they have the same basic meaning, can be mapped to a single word or symbol. Also, in the context of machine learning-based NLP, stemming makes your training data simpler by reducing the number of words to a large extent. This makes the analysis much easier. This makes stemming a very important process in the context of text analysis. This reduces the complexity of text analysis to a great extent and makes the process of text analysis much easier, simpler, and less complex for data analysts. Finding Polarity and Visualization using WordCloud library:

In sentiment analysis, polarity refers to recognizing positive, neutral, and negative sentiment orientation in written or spoken language. The most important part of sentiment analysis is to examine a body of text to comprehend the context and opinions expressed therein. Because computers can only read numerical data, we can use polarity to quantify positive and negative emotions. From the sign of calculated polarity score, the overall sentiment is inferred as positive, neutral, and negative from the user reviews.

WordCloud library in Python is a visual representation of text data. From this visual representation, we can quick valuable insights about whether a majority of users have given positive or negative reviews. The importance of each word is shown by font size or color in this visualization, which consists of words that are usually single words. From this visualization, we got valuable insights about the overall user reviews.

4.3 Advanced Analysis in GCP

This type of analysis is done by using Natural Language Processing (NLP) API in Google Cloud Platform. This API reveals the structure and meaning of text which is used to extract information about text documents, blogs, places, people and events, etc., and analyze the sentiment of the text and parse intent from customer conversions in messages. Some of the few challenges encountered during analysis of static data are

- Data collected may sometimes become bad data which means data that is out-of-date, incomplete, or incorrect in some capacity. So, implementing a more dynamic approach to data collection and data maintenance will significantly improve the quality of your data.
- Also, dynamic data is more readily available and easily accessible as it is readily available on the Internet in various social media platforms like Facebook, Twitter, LinkedIn, and Instagram where people usually raise their views, opinions, and concerns. So, such social media platforms can be readily used to get access to streaming data on the Internet. We can be guaranteed that data in such platforms will never go outdated because most social media platforms give access only to recent data during data mining. They do not allow access to incredibly old data.

4.4 Twitter Analysis

Here, we will be collecting tweets from Twitter in order to analyze the experience of students in virtual learning platforms. Twitter API is used to collect data from Twitter. Here, we have used TextBlob which is a Natural Language Processing library for Python. We have analyzed the tweets of individual tweets in terms of polarity. Also, a small visualization using histogram is done. Steps followed for analysis:

- Setting up the environment: Initially, a twitter developer account is created to get access to all Twitter data. Then, an application is created in the developer account and the secret keys (consumer keys and access token keys) are generated which has to be used in our project to get access to all individual tweets.
- Importing the necessary libraries: Then, all the necessary python libraries like Tweepy, NLTK, textBlob, Pandas, and NumPy are imported which are used in various stages of the analysis. NLTK can be used for various text-processing applications like sentiment analysis. TextBlob is a Python library which is built upon NLTK. It is used in a variety of applications like art-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more.
- Searching the requirement tweets using Twitter API: Now using the secret keys generated, we have searched for tweets. For this, a special method called Cursor() is used. In this method, we give parameters such as the search item, the start date of the search, and the language of the tweets. Also, twitter allows access to only recent tweets. It does not allow access to very old tweets.
- Data Cleaning on Tweets: In every data analysis application, the data must be clean for efficient analysis of data. So, initially, all the noisy data present in the Twitter dataset has to be processed and removed. So, data cleaning plays a very important role in any data analysis application. Then, data cleaning is performed on these tweets extracted. In this process, all numbers, very small words with just one or two characters and all the URLs are removed because all the things do not contribute much to the analysis of data.
- Sentiment Analysis on Tweets: Then, the Pythonpackage textBlob is used to calculate the polarity values of individual tweets on the efficiency of virtual learning platforms. We have created textBlob objects which assign polarity values to the tweets. Then, the polarity values of tweets are identified by using the attribute polarity of textBlobobject [12]. The sentiment property returns a tuple of the form-Sentiment (polarity, subjectivity). The polarity values tell us the sentiment score of the particular text. The polarity score is a float within the range [−1.0, 1.0]. The value of 1.0 indicates positive sentiment, a value near to −1.0 indicates negative sentiment, and a value near to 0.0 indicate neutral sentiment. The subjectivity score is a measure of subjectiveness. The subjectivity is a float within the range [0.0, 1.0] where 0.0 is very objective and is very subjective.
- Sentiment Analysis using Histogram: Finally, a small visualization using a histogram is done to find out the majority of sentiments in user reviews. This is done by using a Python library called Matplotlib. It is an open source alternative for MATPLAB for plotting graphs. This library is mainly used to create static

and interactive plots and visualizations. From this histogram visual, a majority of tweets was found to have a sentiment score of near to 0.0. So, from this it is understood that most of the users had a neutral experience in virtual learning platforms. Thus, graphs like histogram give a quick visualization of the overall reviews given by the users. From the Python data analysis done for the data which was collected from the survey, it was inferred that most of the students found social networking sites and other entertainment sites on the Internet to be very distracting during their online classes. So, in order to provide a simple solution to this, a small chrome extension called "Simple Toggle Sites" was built in order to restrict access to such sites. 1. A chrome extension called "Simple Toggle Sites" was created. When this extension is activated, it would block certain web sites whose sole purpose is for entertainment. This is done in order to avoid any distractions the students may face during online classes. Technologies used for creation are HTML and CSS-Front-end and Javascript for Back-end. Using Javascript, the web sites mentioned in the list will be blocked when the toggle is activated. This is done by creating a listener before every web request, and it will check whether that web site is in the list of webpages mentioned in the list.

5 Results and Discussion

See Table 1.

Using the WordCloud package, we will be able to generate an image that gives us the most representative words in a chosen set of reviews. Some popular words that can be observed above include "fine," "great," "good," and "learn." These words mostly give positive meaning [13, 14] (Tables 2 and 3; Figs. 2, 3, 4, and 5).

Table 1 Analysis of positive sentiments using python

Awesome learning	Positive
Great to access learning stuff	Positive
Awesome platform	Positive
Excellent way to teaching and understanding	Positive

Table 2 Analysis of negative sentiments using python

Too boring and distracting	Negative
Worst and too distracted	Negative
Too hectic and boring	Negative
Too bad experience	Negative
Worst	Negative

Table 3 Sentiment analysis

S. No.	Polarity	Tweet
0	0.000000	RT Bell Foundation Are you for ways to..
1	0.216667	Texas schools urge quick legislative actions so..
2	−0.300000	The COVID19 pandemic reshaped the education la..
3	0.000000	A virtual COVID19 Response Town Hall Monday 52..
4	−0.062500	This is my 3rd year teaching and i am still le..
5	0.000000	We are back to school virtually tomorrow we ar..
6	0.000000	Virtual learning lab COVID19 vaccine in the Wor..
7	0.450000	BT Brizard JC the Centre for inclusive innovation..
8	0.268182	RT ADHR Journal Two new ADHR Journal articles a…
9	0.268182	Two new ADHR journal articles are available Onl..
10	0.033333	12 The impact of COVID19 Promoted virtual remote..

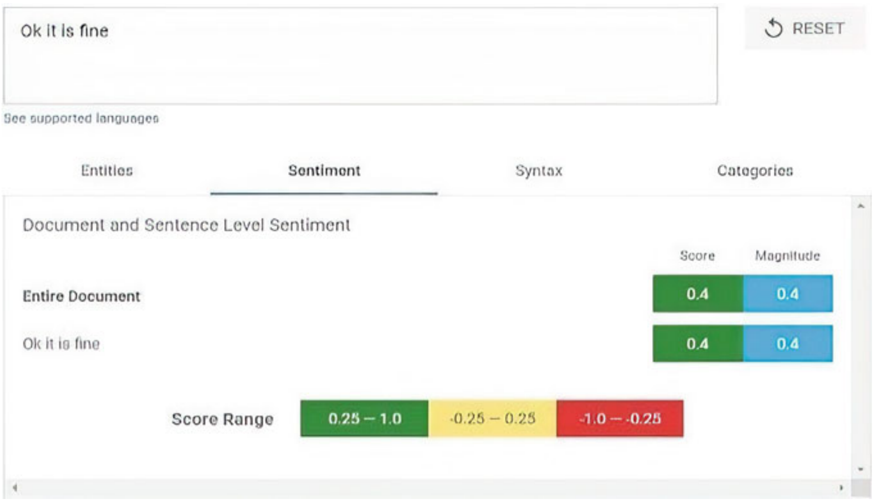


Fig. 2 Result from sentiment analysis (GCP)

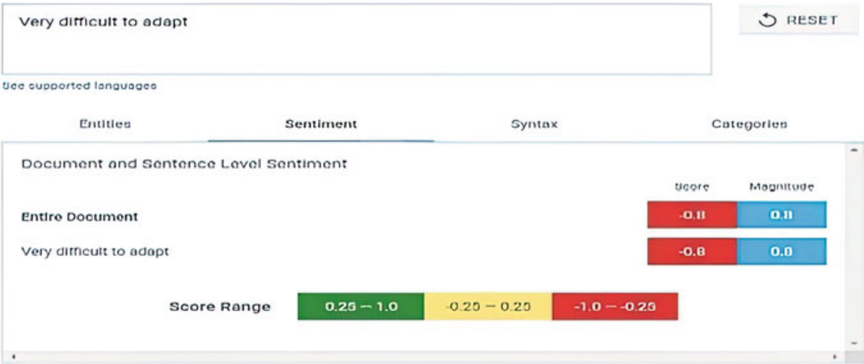


Fig. 3 Twitter data analysis

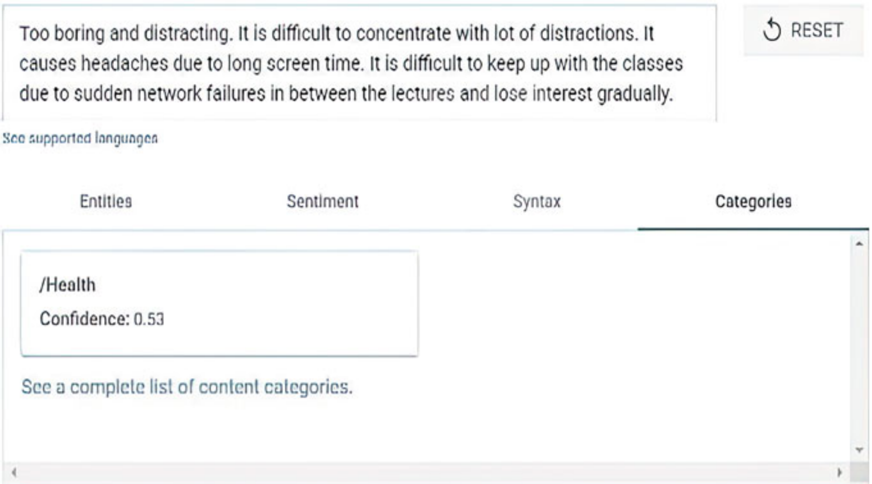


Fig. 4 Result from sentiment analysis (GCP)

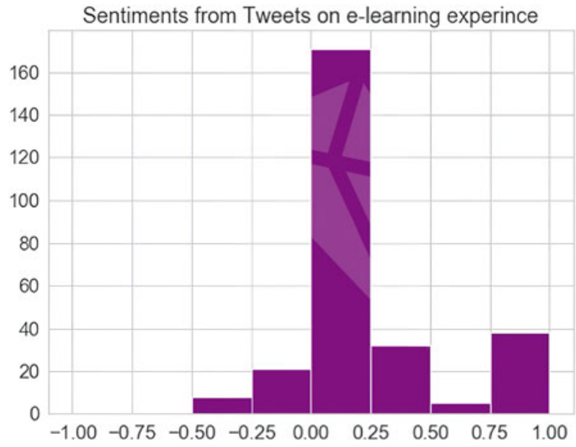


Fig. 5 Sentiment analysis from tweets

6 Conclusion and Future Enhancement

Thus, in this study we have accurately analyzed various concerns faced by students in online learning platforms. Also, we provide certain recommendations and solutions to enhance the overall learning experience of the students. In the future, we can get data from various social networking sites like Facebook and LinkedIn so that we can get reviews from many users. Also, we can derive data about the number of likes, retweets, and comments, etc., and analyze that. These will significantly increase the efficiency of analysis and will enable us to accurately analyze the data and provide better solutions and recommendations. We can also combine Adaptive Neural Network for the target-dependent Twitter sentiment classification in future [15, 16].

References

1. Jena PK (2020) Impact of covid-19 on higher education in India. Int J Adv Educ Res (IJAER) 5
2. Chen T, Peng L, Yin X, Rong J, Yang J, Cong G (2020) Analysis of user satisfaction with online education platforms in China during the COVID-19 pandemic. Healthcare 8(3):200. https://doi.org/10.3390/healthcare8030200
3. Pozgaj Z, Knezevic B (2007) E-learning: survey on students' opinions. 29th International Conference on Information Technology Interfaces. https://doi.org/10.1109/iti.2007.4283800
4. Yue L, Chen W, Li X, Zuo W, Yin M (2019) A survey of sentiment analysis in social media. Knowl Inf Syst 60:617–663. https://doi.org/10.1007/s10115-018-1236-4
5. Vishal A, Kharde S, Sonawane S (2016) Sentiment analysis of twitter data: a survey of techniques. Int J Comput Appl 139(11):5–15. https://doi.org/10.5120/ijca2016908625
6. Radhika G, Narasimha Prasath K, Rajasundari (2018) Sentimental analysis on Tamil news feed. J Adv Res Dyn Control Syst 9(Special issue 11):397–399
7. Neethu MR, Harini N, Abirami K (2021) Investigating peers in social networks: reliable or unreilable. In: Lecture notes in networks and systems, vol 127, pp 173–180

8. Jayan AP, Balasubramani A, Kaikottil A, Harini N (2019) An enhanced scheme for authentication using OTP and QR code for MQTT protocol. Int J Recent Technol Eng 7(5):70–75
9. Veeramanickam MRMA, Radhika N (2014) A smart E-Learning System for social networking. Int J Electr Comput Eng Open Access 4(3):447–4551
10. Shriya S, Vinayakumar R, Anand KM, Soman KP (2016) Predicting the sentimental reviews in Tamil movie using machine learning algorithms. Indian J Sci Technol 9(45)
11. Kumar A, Kohail S, Ekbal A, Biemann C (2015) IIT-TUDA: System for sentiment analysis in Indian languages using lexical acquisition. In: Mining intelligence and knowledge exploration, pp 684–693
12. Liao J, Wang S, Li D, Li X (2017) FREERL: fusion relation embedded representation learning framework for aspect extraction. Knowl Based Syst 135:9–17
13. Deng L, Wiebe J (2016) Recognizing opinion sources based on a new categorization of opinion types. IJCAI 2016:2775–2781
14. Jin Y (2017) Development of word cloud generator software based on python. Procedia Eng 174:788–792
15. Dong L, Wei F, Tan C, Tang D, Xu K (2014) Adaptive recursive neural network for target-dependent twitter sentiment classification. In: Meeting of the association for computational linguistics
16. Li Y, Li X, Yu G (2015) Research of sentiment classification based on the Chinese stock blog. J Wuhan Univ (Natural Science Edition) 61(2):163–168

Analysis of Browsing Activity of Portable Opera Browser in Windows 10 Pro System in VMware Workstation Using Digital Forensics Software

Arjun Chetry and Uzzal Sharma

Abstract For human beings, irrespective of their profession, Internet connectivity became a necessity for survival in this digital age, but at the same time, privacy is becoming a growing concern for every Internet user. With rapid growing trends of personalized Internet activity, digital profiling is becoming a new normal, and all Internet servers are compiling users' activities to provide better suggestions or recommendations. So, to avoid compilation of such activities, users are using facilities like portable browsers, private browsers, and tor browsers to remain anonymous. When similar facilities are used by criminals, then it is becoming a challenge for law enforcement agencies to investigate cases. Therefore, in this paper, we intend to analyze the portable browser in windows 10 pro in virtual machine for exploring the possible artifacts retrieval using digital forensics software. During analysis, it was found that various types of artifacts were available about the browsing activity from virtual machine hard drive and from RAM dump.

Keywords Digital evidence · Live forensics · Memory forensics · Portable browser · Volatility framework · FTK · Opera browser

1 Introduction

Digital profiling in Internet for better personalized service is becoming new normal. However, from the point of view of personal privacy, users are avoiding such monitoring of Internet activities by using portable browser or private mode [1]. It is pertinent to mention that even the criminals prefer to use such mode of mode browsers to avoid tracking their Internet activities or to make themselves anonymous. Modern day browsers are having private browsing mode to protect user's privacy, but cybercriminals are using these facilities to remain anonymous while executing various types of crimes [2]. Therefore, it is vital for law enforcement agencies to find out various artifacts available in the system, whenever there is any crime committed using

A. Chetry (✉) · U. Sharma
Assam Don Bosco University, Guwahati, India
e-mail: chetry.arjun@gmail.com

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_48

such private browsing mode. In this paper, it is intended to perform forensics analysis of system to check whether portable modern browsers are collecting or saving any types of data or not. So, simulation scenario is designed using virtual machine, and some activities are performed in portable browsers in private mode followed by forensics analysis to retrieve evidences related to these activities as performed similarly for portable browsers in [3, 4].

2 Modern Day Browsers and Anonymity Techniques

With time, many browsers are launched in the Internet for privacy of Internet users, and most of them are available to execute in portable mode. Some of the example of portable modern day browsers are as under:

i. Opera portable browser
ii. Falkon portable Web browser
iii. Google Chrome portable (Freeware)
iv. Mozilla Firefox, portable edition
v. Iron portable—advanced Web browser.

Out of these browsers, **Opera portable browser** is selected for this paper purpose, and forensics analysis is performed to understand the recoverable artifacts from Opera portable browser.

3 Experimental Setup for Simulation

It is observed that many techniques are used by cyber-criminals to avoid leaving any evidences in the system. So, to create the simulation environment, virtual machine setup is used, and forensically, clean media is engaged for IR tools and system. Criminal may use USB stick or memory card to store the portable browsers and run the application in portable mode, so the similar setup is used for this experiment [5]. To collect the evidences, forensically, clean USB storage device is used, and VMDK file, RAM dump, and related evidences were collected for the analysis as discussed below.

i. **Windows 10 Pro Setup**:
   Windows 10 pro is set up on VMware workstation 16 as shown in Fig. 1. This system is a fresh system and not having any previous residual data in

this system. Therefore, it is presumed that the initial registry is fresh, and no previous browsing history is available in this system.

ii. **Preparation of USB drive with portable Web browsers**:
Many platforms or modern-day browsers available for downloading of portable browsers. For this experiment, portableApps Web site [6] is accessed for downloading of portable browsers, and accordingly, it is installed on forensically clean USB drive as shown in Fig. 2.

iii. **Criminal activity performed on System**:

Suppose the system was used by anonymous user to perform criminal activities using portable opera browser and, for example, the accused was accessing various Web sites in the Internet like : www.ahmia.fi, www.google.com, www.facebook.com, www.gmail.com, www.cybernews.com, www.youtube.com, and many more. Using NirSoft tools [7], activity performed in the system is recorded as shown in Figs. 3 and 4.

iv. **Digital Forensics Tools on USB Drive**:

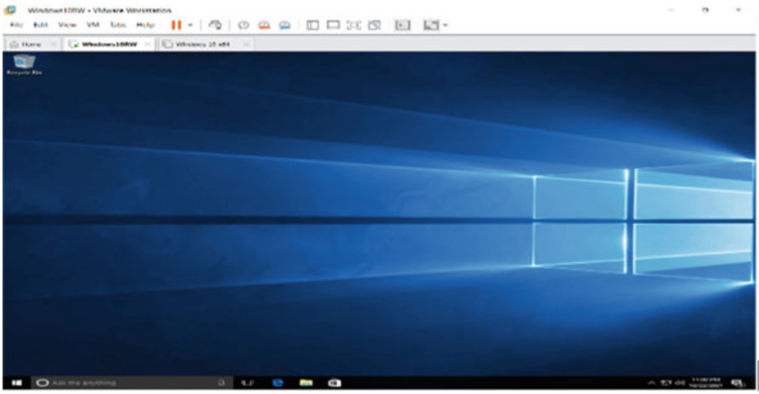


Fig. 1 Windows 10 pro on VMware workstation

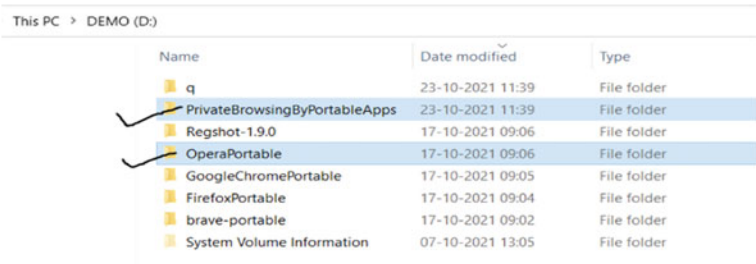


Fig. 2 Portable browsers installation on USB drive

Fig. 3 Details of last activity performed in the system

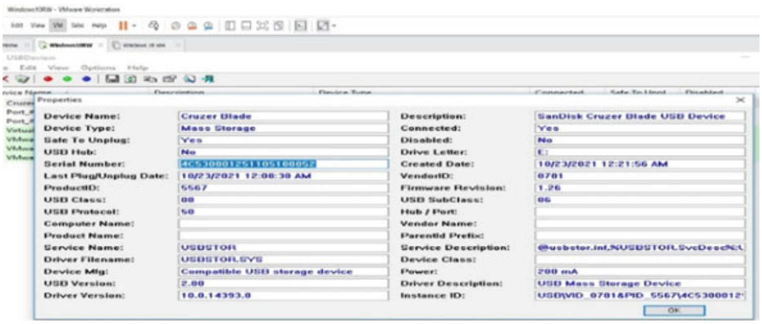


Fig. 4 Details of USB connected in the system

It is pertinent to mention that privacy is important for every Internet user, but at the same time, if device is used by anonymous criminal for committing crime, then it is important to know what are artifacts are possible to retrieve from such system and RAM during investigation. Therefore, in this paper, we intend to find out the different types of artifacts from system drive, from RAM of system and from USB drive. Details of tools engaged during experiment are as under:

Stages	Steps performed
Ftk Imager	Ftk Imager was used for capturing the RAM data as well as to image the storage drive of this system into external hard disk drive for analysis
Regshot	Registry tool was engaged to compare the registry artifacts involved before and after executing the portable browser in the system
Magnet Axiom Forensics Software and Autopsy forensics tool	Magnet Axiom forensics software and autopsy forensics tool were used for analysis of digital evidence to find out all relevant artifacts

(continued)

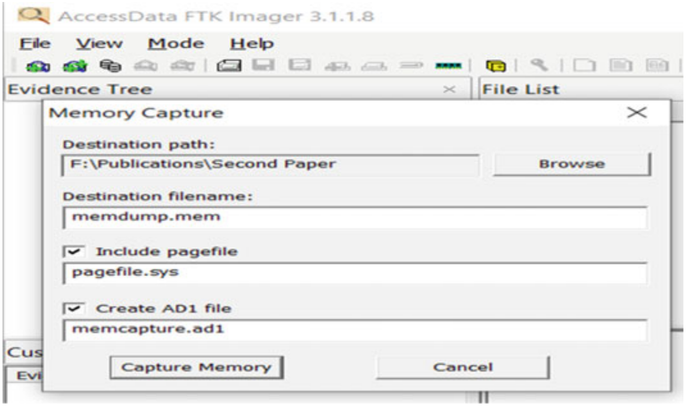


Fig. 5 Imaging of RAM and pagefile.sys

(continued)

Stages	Steps performed
Volatility framework	Command line volatility tool was used for analysis of memory artifacts from captured memory of this system to list the live process details and other related evidences

4 Acquisition of Digital Evidence

Regshot software is used to capture the registry of the system as registry stores all the artifacts of windows system [8]. Thereafter, volatile data from RAM were collected using Ftk Imager tool for analyzing the live data of RAM as shown in Fig. 5.

Similarly, Ftk Imager was used for imaging of storage media, VMDK file of virtual machine and saved in USB hard disk drive.

5 Digital Forensics Analysis and Results

During analysis of VMDK file, Magnet Axiom software was used, and various artifacts related to Internet activity performed in this system are retrieved as shown in figure. This shows the instance of opera.exe was running in this system from IR tools folder of pendrive (Figs. 6, 7, 8, 9 and 10).

During analysis, it was also found that there are URLs list available in system for Web sites visited from this system. Also, it is found that the SMTP.Gmail.com URL is available during analysis of vmdk file as one test email was sent from portable browser

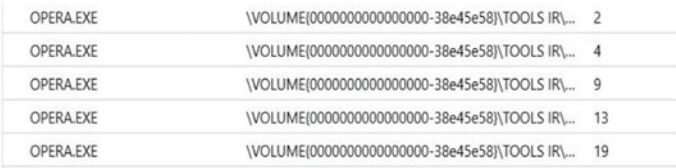


Fig. 6 Details of opera process execution—from VMDK file

Fig. 7 Opera portable browser executed from USB drive—from VMDK file

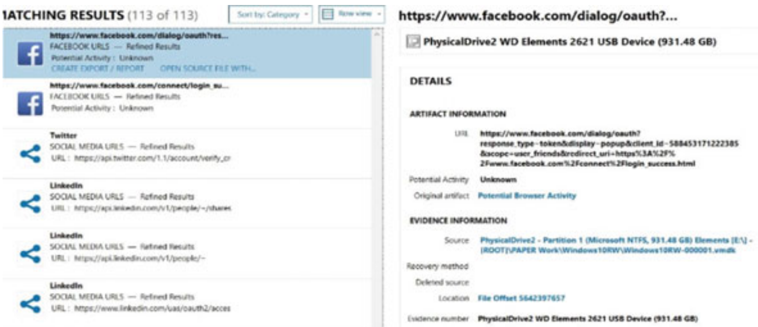


Fig. 8 Various URLs accessed in this system—from VMDK file

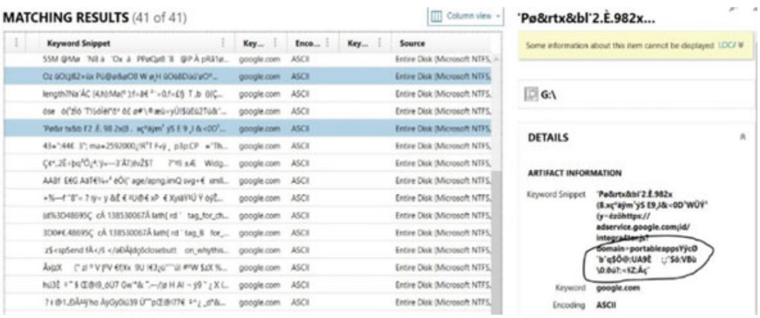


Fig. 9 PortableApps information—retrieved from VMDK file

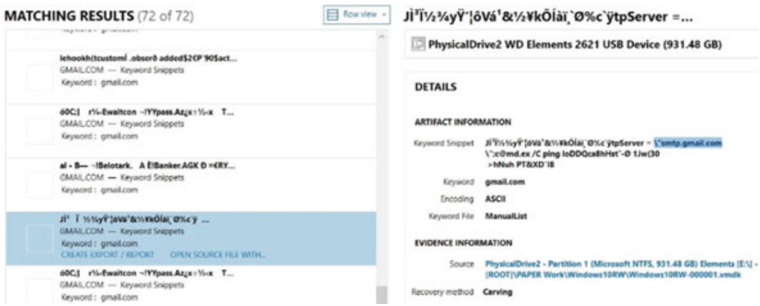


Fig. 10 Gmail SMTP protocol information—retrieved from VMDK file

during simulation. Also, Magnet Axiom shows the statistics of various artifacts links available during analysis. All downloaded files were stored in USB drive during simulation exercise and during analysis; all those downloaded links and storage location are available on VMDK file (Figs. 11 and 12).

From the results of analysis of VMDK file of virtual machine installed with windows 10 pro, it is summarized that there are multiple files or information like Web site visited, downloaded, email accessed, and opera processes are possible to retrieve from VMDK file of the system. This shows that the information about portable browser and related information is recoverable from the system [9]. Even

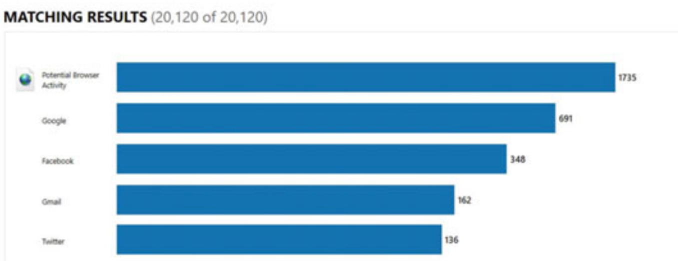


Fig. 11 Multiple information about Web activity—from VMDK file

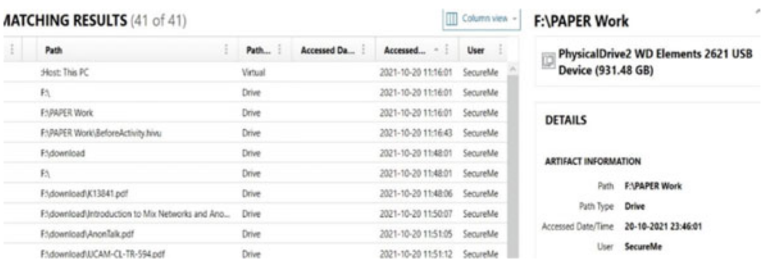
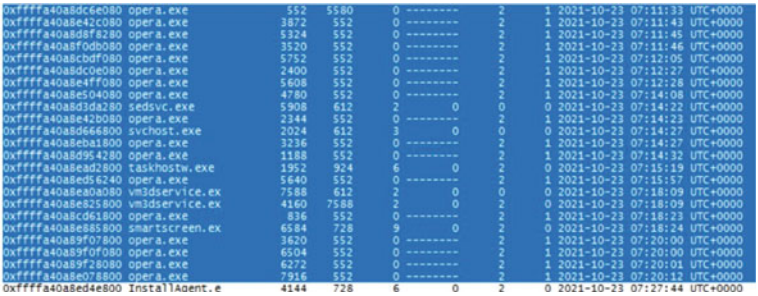


Fig. 12 Downloaded and saved file information—retrieved from VMDK file



```
0xffffa40a8dc6e080 opera.exe        552  5580  0 ---------  2  1 2021-10-23 07:11:33 UTC+0000
0xffffa40a8e42c080 opera.exe       3872   552  0 ---------  2  1 2021-10-23 07:11:43 UTC+0000
0xffffa40a8d8f8280 opera.exe       5324   552  0 ---------  2  1 2021-10-23 07:11:45 UTC+0000
0xffffa40a8f0db080 opera.exe       3520   552  0 ---------  2  1 2021-10-23 07:11:46 UTC+0000
0xffffa40a8cbdf080 opera.exe       5752   552  0 ---------  2  1 2021-10-23 07:12:05 UTC+0000
0xffffa40a8dc0e080 opera.exe       2400   552  0 ---------  2  1 2021-10-23 07:12:27 UTC+0000
0xffffa40a8e4ff080 opera.exe       5608   552  0 ---------  2  1 2021-10-23 07:12:28 UTC+0000
0xffffa40a8e504080 opera.exe       4780   552  0 ---------  2  1 2021-10-23 07:14:08 UTC+0000
0xffffa40a8d3da280 sedsvc.exe      5908   612  2         0  0  0 2021-10-23 07:14:22 UTC+0000
0xffffa40a8e42b080 opera.exe       2344   552  0 ---------  2  1 2021-10-23 07:14:23 UTC+0000
0xffffa40a8d666800 svchost.exe     2024   612  3         0  0  0 2021-10-23 07:14:27 UTC+0000
0xffffa40a8eba1800 opera.exe       3236   552  0 ---------  2  1 2021-10-23 07:14:27 UTC+0000
0xffffa40a8d954280 opera.exe       1188   552  0 ---------  2  1 2021-10-23 07:14:32 UTC+0000
0xffffa40a8ead2800 taskhostw.exe   1952   924  6         0  2  0 2021-10-23 07:15:19 UTC+0000
0xffffa40a8ed56240 opera.exe       5640   552  0 ---------  2  1 2021-10-23 07:15:57 UTC+0000
0xffffa40a8ea0a080 vm3dservice.ex  7588   612  2         0  0  0 2021-10-23 07:18:09 UTC+0000
0xffffa40a8e825800 vm3dservice.ex  4160  7588  2         0  2  0 2021-10-23 07:18:09 UTC+0000
0xffffa40a8cd61800 opera.exe        836   552  0 ---------  2  1 2021-10-23 07:18:23 UTC+0000
0xffffa40a8e885800 smartscreen.ex  6584   728  9         0  2  0 2021-10-23 07:18:24 UTC+0000
0xffffa40a89f07800 opera.exe       3620   552  0 ---------  2  1 2021-10-23 07:20:00 UTC+0000
0xffffa40a89f0f080 opera.exe       6504   552  0 ---------  2  1 2021-10-23 07:20:00 UTC+0000
0xffffa40a89f28080 opera.exe       6272   552  0 ---------  2  1 2021-10-23 07:20:01 UTC+0000
0xffffa40a8e078800 opera.exe       7916   552  0 ---------  2  1 2021-10-23 07:20:12 UTC+0000
0xffffa40a8ed4e800 InstallAgent.e  4144   728  6         0  2  0 2021-10-23 07:27:44 UTC+0000
```

Fig. 13 Process list of opera portable browser—from RAM dump

with the help of RAM dump analysis [10], many information about the running processes in the system and the list of opera running process is found as shown in Fig. 13. Therefore, it is possible for law enforcement agencies or investigator to retrieve artifacts which may help investigator in proving the case to the court of law.

6 Conclusion and Future Work

As mentioned above, various types of information are available in system drive and in RAM dump about the activities performed by accused using portable browser. Therefore, it is concluded that the portable browsers are leaving various types of information in the system which may be used by investigator during investigation. In future work, further analysis may be performed for multiple portable browsers to find out the list of evidences possible to retrieve from portable browsers, especially if it executed in private mode.

References

1. Choi, J. H., Lee, K. G., Park, J., Lee, C., & Lee, S.: Analysis framework to detect artifacts of portable web browser. Information Technology Convergence, Secure and Trust Computing, and Data Management, 2012, 207–214.
2. Dave R, Mistry NR, Dahiya MS (2014) Volatile memory based forensic artifacts and analysis. Int. J. Res. Appl. Sci. Eng. Technol. 2(1):120–124
3. Chivers H (2014) Private browsing: a window of forensic opportunity. Digit Investig 11(1):20–29
4. Umar, Rusydi & Yudhana, Anton & Faiz, Muhammad. (2018). Experimental Analysis of Web Browser Sessions Using Live Forensics Method. International Journal of Electrical and Computer Engineering (IJECE). 8. 2951. https://doi.org/10.11591/ijece.v8i5.pp 2951–2958.
5. Smith, Cory & Dietrich, Glenn & Choo, Kim-Kwang Raymond. (2018). Identification of Forensic Artifacts in VMWare Virtualized Computing. https://doi.org/10.1007/978-3-319-78816-6_7.
6. https://portableapps.com/apps/internet, accessed on 23/10/2021

7. https://www.nirsoft.net/ accessed on 23/10/2021
8. Duranec, A. & Gruicic, Savina & Zagar, Marinko. (2020). Forensic analysis of Windows 10 Sandbox. 1224–1229. https://doi.org/10.23919/MIPRO48935.2020.9245226.
9. Hasan, Fayyad-Kazan & Sondos, Kassem-Moussa & Hejase, Hussin & Hejase, Ale. (2021). Forensic analysis of private browsing mechanisms: Tracing internet activities. Journal of Forensic Science and Research. 5. 012–019. https://doi.org/10.29328/journal.jfsr.1001022.
10. Chetry A., Sharma U. (2021) Dark web Activity on Tor—Investigation Challenges and Retrieval of Memory Artifacts. In: Gupta D., Khanna A., Bhattacharyya S., Hassanien A.E., Anand S., Jaiswal A. (eds) International Conference on Innovative Computing and Communications. Advances in Intelligent Systems and Computing, vol 1165. Springer, Singapore. https://doi.org/10.1007/978-981-15-5113-0_80

Breast Cancer Diagnosis Using Histopathology and Convolution Neural Network CNN Method

Mazhar B. Tayel, Mohamed-Amr A. Mokhtar, and Ahmed F. Kishk

Abstract High-performance computer tools have been more widely available, and deep learning systems that utilize deep neural networks have become increasingly common in many fields. Deep learning approaches based on convolution neural networks (CNN) have become more widespread as high-performance computer facilities have grown. An overview of the growth of deep learning models and a concise explanation of various learning approaches, such as supervised learning, trains the neural network using labeled data. Solid experiments are required in medical image analysis studies to prove the efficacy of proposed approaches. Many architectures, such as Pre-trained Networks and Convolution Neural Networks CNN, are employed to achieve breast cancer diagnosis. Various classification measures may be utilized, making comparison of the methodologies challenging. Medical screening methods have grown increasingly important in the detection and treatment of diseases. Early identification of breast cancer is regarded to be a crucial element in lowering women's mortality rates. Several different breast screening modalities are being investigated to improve breast cancer diagnosis. Histopathology is used in a current cancer detection and localization method that uses artificial intelligence to screen for breast cancer and identify the existence of tumors in the breast. This study focused on an experimental dataset that employed convolution neural network (CNN) techniques to detect and localize breast tumors (i.e., pre-trained CNN). CNNs are a powerful tool for solving real-world problems, and neural networks with learning algorithms are a promising new technology.

Keywords Breast tumor (BT) · Convolution neural network (CNN) · Histopathology · Deep learning (DL) · Artificial neural network (ANN)

M. B. Tayel
EED, Faculty of Engineering, Alexandria University, Alexandria, Egypt

M.-A. A. Mokhtar · A. F. Kishk (✉)
EED, Alexandria Higher Institute of Engineering and Technology (AIET), Alexandria, Egypt
e-mail: ahmed.kishk@aiet.edu.eg

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_49

1 Introduction

Breast tumor (BT) is a major public health issue in the modern world. Breast tumors (BT) are harmful to women and can be fatal if not treated early. Mammography, Computer-Aided Detection (CAD), Ultrasound Imaging, Magnetic Resonance Imaging (MRI), Positron Emission Tomography (PET), Optical Imaging and Spectroscopy (OIS), Thermography, Electrical Impedance Imaging, Electronic Palpation, and Electrical Potential Measurements are all examples of breast cancer screening methods. It is a necessary aspect of the diagnosis, treatment, and prognosis for most breast diseases. Due to the increasing growth in breast cancer, tools such as breast self-examination (BSE), imaging, surgery, chemotherapy, radiation treatment, palliative care, and cancer education are all used for early cancer diagnosis and management. Biopsy, on the contrary, is the only means to know for sure if cancer is present. Fine needle aspiration, core needle biopsy, vacuum-assisted, and open biopsy are the most common surgical biopsy procedures (SOB). The method entails collecting cells or tissue samples, which are then fixed across a glass microscope slide for staining and microscopic analysis [1]. The most typical treatment for breast cancer is surgical removal of the breasts, followed by chemotherapy and radiation therapy. Tumor size, lymph node status, histological type, histological grade, hormone receptor presence or absence, and patient age all play a role in chemotherapy and radiation therapy for breast cancer [2]. The database is made up of clinically realistic microscopic images of breast tumor tissue taken from 82 patients using various magnification factors (40) as given in [1]. Microscopic biopsy images of benign and malignant breast cancers are available in this database. Breast tissue biopsy slides stained with hematoxylin and eosin are used to create samples case of study. Pathologists from the P&D Lab collect the samples via surgical (open) biopsy (SOB), prepare them for histological analysis, and label them. The preparation of the procedure includes steps and the acquisition system; also, the format and dimension of the captures images are mentioned in [1]. The database is in dimension of 700×460 pixels, and hence, the dimension of the image is then adjusted to be entered to different CNN architecture for the purpose of performance comparison. The database consists of the following cases as given in Table 1. Also, database in the reference [1] is given in portable network graphic (PNG) format which have a size on disk 1.08 GB before using data augmentation which is too large, hence changing the format of the database to be small size on disk using Joint Photographic Experts Group (JPG) format to compress the database to 152 MB after using data augmentation see Table 2.

A critical step in evaluating a breast cancer is testing a piece of the cancer removed during the biopsy (or surgery) to discover whether it possesses estrogen and progesterone receptors [3].

Breast lesions are classified as follows [3]:

(a) Benign adenosis: A benign adenosis is a gland sickness or abnormal change. Breast adenosis is a benign condition in which the lobules of the breast are larger than they should be as shown in Fig. 1a.

Table 1 Benign and malignant classes' number of cases after data augmentation on original database

Benign	Number of cases before data augmentation	Number of cases after data augmentation	Malignant	Number of cases before data augmentation	Number of cases after data augmentation
Adenosis	114	684	Ductal carcinoma	864	5184
Fibroadenoma	253	1518	Lobular carcinoma	156	936
Phyllodes tumor	109	654	Mucinous carcinoma	205	1230
Tubular adenoma	149	894	Papillary carcinoma	145	870

(b) Ductal carcinoma (malignant): The cancer has spread outside of the breast duct to normal tissue [4] as shown in Fig. 1b.

(c) Fibroadenoma (Benign): Fibroadenoma is a benign tumor made up of glandular and stromal (connective) tissue. Because fibroadenomas grow during puberty, they are most found in young women, but they can afflict women of any age. Fibroadenomas can also affect men; however, they are uncommon as shown in Fig. 1c.

(d) Lobular carcinoma (malignant): Lobular carcinoma is a rare condition in which abnormal cells form in the breast milk glands (lobules) [5] as shown in Fig. 1d.

(e) Mucinous carcinoma (malignant): Mucinous carcinoma is a kind of breast cancer marked by the presence of extracellular mucin and is associated with a better prognosis than any other type of invasive breast carcinoma [6] as shown in Fig. 1e.

(f) Papillary carcinoma (malignant) is an epithelial tumor with follicular cell differentiation and a distinctive set of nuclear features that can affect any part of the ductal tree [7] as shown in Fig. 1f.

(g) Phyllodes tumor (Benign): Phyllodes tumors are fibroepithelial lesions that are uncommon. Correct surgical planning and avoidance of reoperation are made possible by accurate preoperative pathology diagnosis [8] as shown in Fig. 1g.

(h) Tubular adenoma (Benign): Tubular adenomas (Benign) are rare benign breast tumors with a restricted mass, robust lobular proliferation, and closely packed small channels with inadequate supporting stroma [9] as shown in Fig. 1h.

2 Types of Breast Cancer

Tumors are masses of aberrant tissues that are classified as either non-cancerous ('benign') or cancerous ('malignant'). Shape, size, stiffness, and viscosity are all

Table 2 Benign and malignant image size after changing format

Benign	Size of cases before data augmentation (MB)	Size of cases after data augmentation (MB)	Compression ratio	Malignant	Size of cases before data augmentation (MB)	Size of cases after data augmentation (MB)	Compression ratio
Adenosis	55.9	8.82	6.33	Ductal carcinoma	472	77.1	6.12
Fibroadenoma	151	22.5	6.71	Lobular carcinoma	77.7	12.1	6.42
Phyllodes tumor	56.6	9.02	6.27	Mucinous carcinoma	117	17.7	6.61
Tubular adenoma	99.4	14.9	6.67	Papillary carcinoma	84.1	13.3	6.32

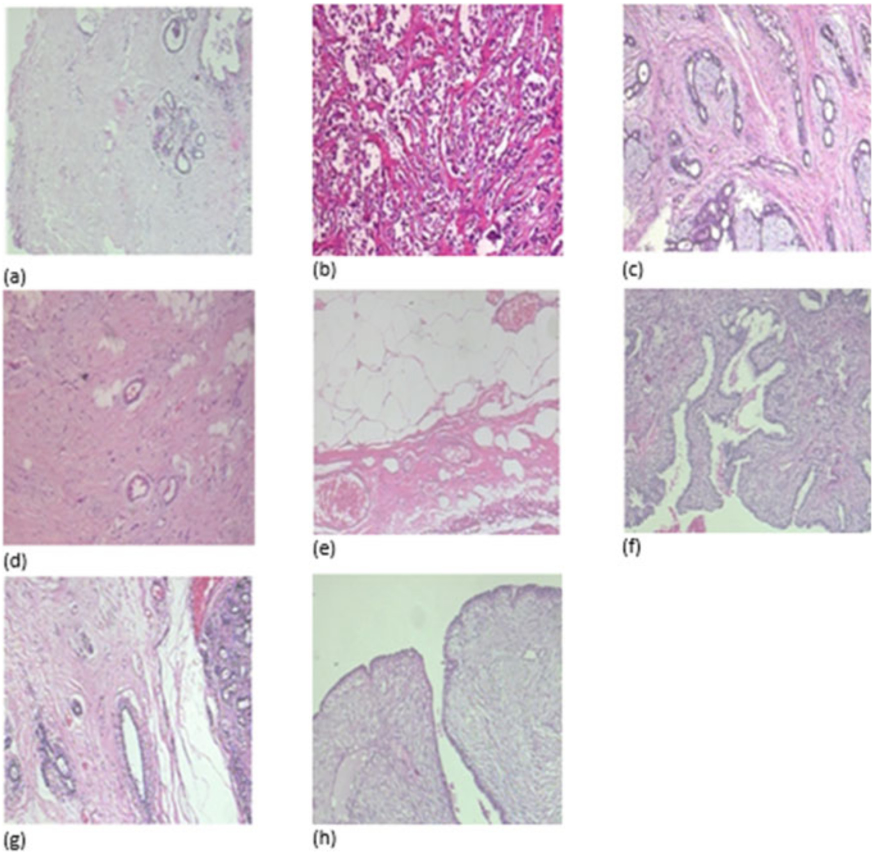


Fig. 1 **a** Adenosis benign, **b** Ductal carcinoma malignant, **c** Fibroadenoma benign case, **d** Lobular carcinoma malignant, **e** Mucinous carcinoma malignant, **f** Papillary carcinoma malignant, **g** Tubular adenoma benign, **h** Phyllodes tumor benign [1]

indicators that can help distinguish between benign and malignant tumors as shown in Table 3.

3 Proposed Methodologies

Histopathology database is gathered and classified into two classes each of four groups: adenosis (Benign), ductal carcinoma (Malignant), fibroadenoma (Benign), lobular carcinoma (Malignant), mucinous carcinoma (Malignant), papillary carcinoma (Malignant), phyllodes tumor (Benign), and phyllodes with magnification factors 40 × [1]. Where the total number of benign databases is 625 images, the total number of malignant images is 1370 of dimension 700 × 460 × 3, see Table

Table 3 Distinction between a benign and a malignant tumor

Benign tumors	Malignant tumors
Do not expand into nearby tissue	Invade the surrounding tissue
They usually do not come back once they have been removed. Magnetic induction	Can reappear after being taken away
It is not possible for it to spread to other sites in the body	It has the potential to spread to other places of the body
If removed, it is unlikely to reoccur or require additional treatment such as radiation or chemotherapy	Surgery, radiation, chemotherapy, and immunotherapy drugs may be required as part of an aggressive treatment plan

1. The suggested method makes use of a total of 11,970 histopathology database after data augmentation methods are used such as image cropping, adding different type of noise to images. The histopathology database is then split into two classes, benign and malignant, each with a different percentage of data. In our example, the data is split into 70% training and 30% validation, totaling 8379 training data and 3591 validation data. The block diagram shown in Fig. 2 represents the steps used for training the given data. The proposed methodology is used with four different CNN architectures to get compare their performance on the given diagnosis cases. The four architectures used are AlexNet, SqueezeNet, GoogLeNet, and ResNet18 with input images dimension 227*227*3 for first two CNN and 224*224*3 for the second two CNN. Hence, the c dimension of the given database in [1] is to be changed to required CNN architecture input dimension as shown in flowchart of Fig. 3. To determine the best number of Mini-Batch Gradient Descent, size is by applying the proposed Eq. (1). The maximum number of epochs in the training progress is given by Eq. (2). The number of iterations per epoch is equal to the number of batches and the validation frequency given by Eq. (4).

$$\text{min} - \text{batch gradient descent}\, m = \sqrt{N} m = \sqrt{N} \tag{1}$$

hence $m = \sqrt{8379} = 96$

where N is the total number of training data.

$$\text{Max epochs}\ E = 2^n \tag{2}$$

where

Fig. 2 The proposed training and testing of the histopathology database are depicted in a block diagram

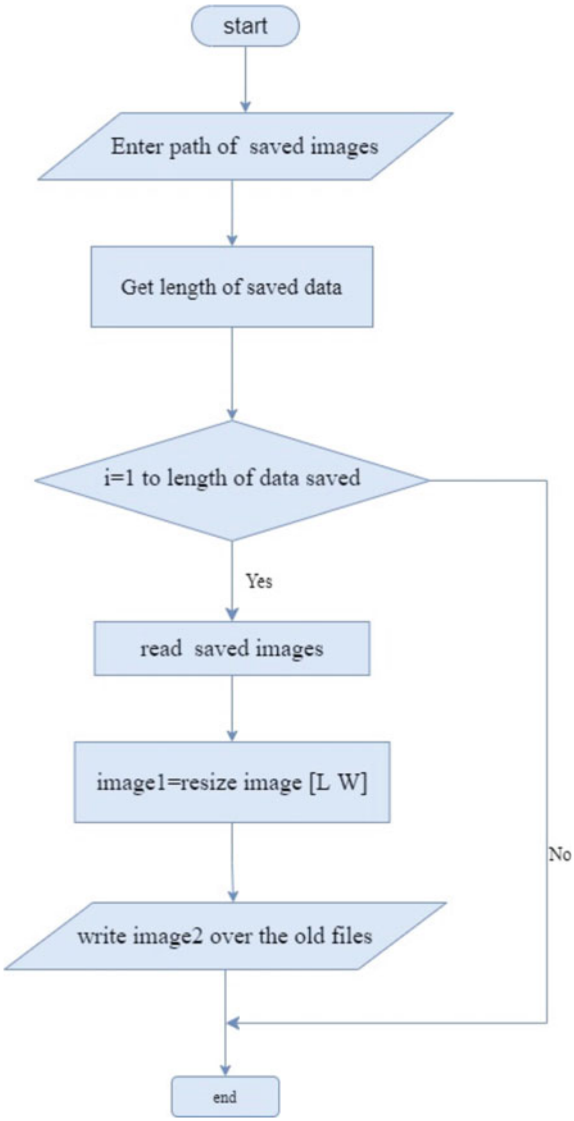


Fig. 3 Flowchart of RGB image (L_ (old value) × W_ (old value) × 3) resize to RGB image (L_ (new value) × W_ (new value) × 3) application system tab algorithm

$$n = \log \sqrt{N} \tag{3}$$

$$n = \log(96) = 1.9 \approx 2$$

$E = 2^2 = 4$ epochs.

Number of batches (NE) = number of iterations (NI) = validation frequency (VF)

$$= \frac{\text{total number of training data}}{\min i \text{ batchs ize}} = \frac{N}{m} = \frac{8379}{96} = 87 \quad (4)$$

3.1 CNN AlexNet Training Progress Labeled Histopathology Data

In this instance, the data is trained using AlexNet, which is 8 layers deep, with an input layer with a dimension of 227 × 227 × 3. Training is done using labeled data then selecting the percentage of training and validation which are 70% and 30%, respectively, see Fig. 4. The training and feature results are shown in Table 4 where the trained AlexNet model has a 97.33% accuracy, and the elapsed time, number of epochs, total number of iterations, iteration per epoch, validation frequency, and mini-batch size are the features measured. The AlexNet training performance is

Fig. 4 Flowchart of training CNN algorithm

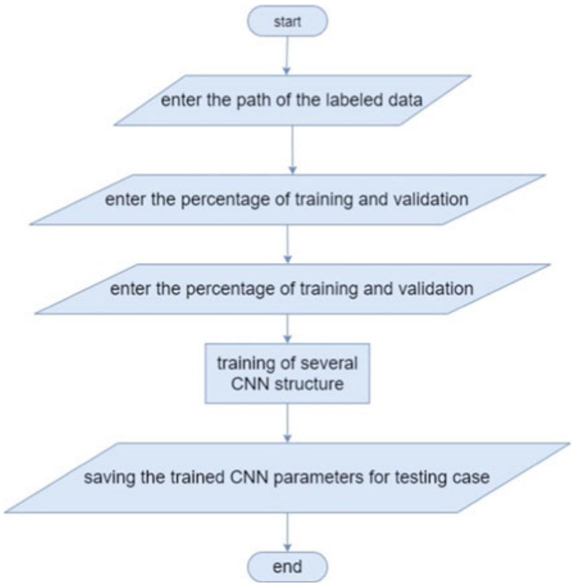


Table 4 Using a histopathology database, AlexNet training accuracy, and training features were determined

Quantity	Value
Accuracy	97.33%
Elapsed time	73 m 2 s 1 h 13 m 2 s
Epochs	4
Iteration	348
Iteration per epoch	87
Validation frequency	87
Mini-batch size	96

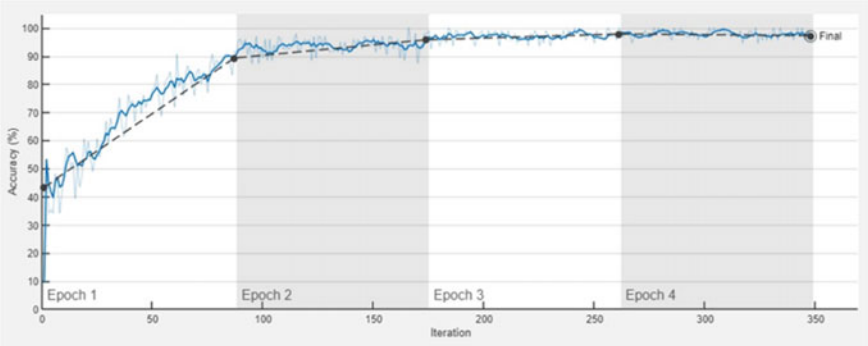


Fig.5 Using 8 histopathological labeled data, the trained AlexNet model's accuracy

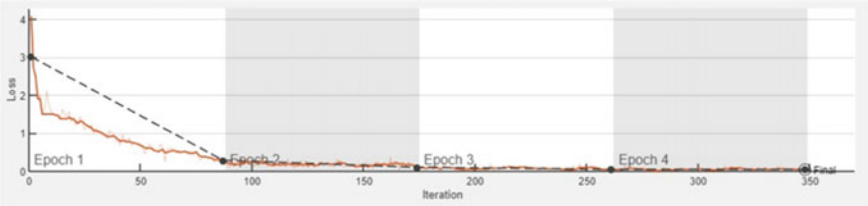


Fig. 6 For trained AlexNet with 8 labeled histopathological data, loss versus iteration

represented by the accuracy vs the number of epochs in the shown Fig. 5. The difference between the AlexNet CNN models predicted value, and the true value is defined as loss and is shown in Fig. 6. The trained AlexNet confusion matrix with histopathological data is shown in Fig. 7 which is used to measure the model efficiency.

3.2 CNN SqueezeNet Training Progress Labeled Histopathological Data

In this procedure, the data is trained using SqueezeNet, which is 18 layers deep, as well as an input layer with a size of 227 × 227 × 3, which contains the training results and features as shown in Table 5. The accuracy versus number of epochs is represented by the SqueezeNet training performance as shown in Fig. 8. The difference between the predicted value by the SqueezeNet CNN model and the true value is termed as loss and is shown in Fig. 9. With histopathological data, the trained SqueezeNet confusion matrix is shown in Fig. 10.

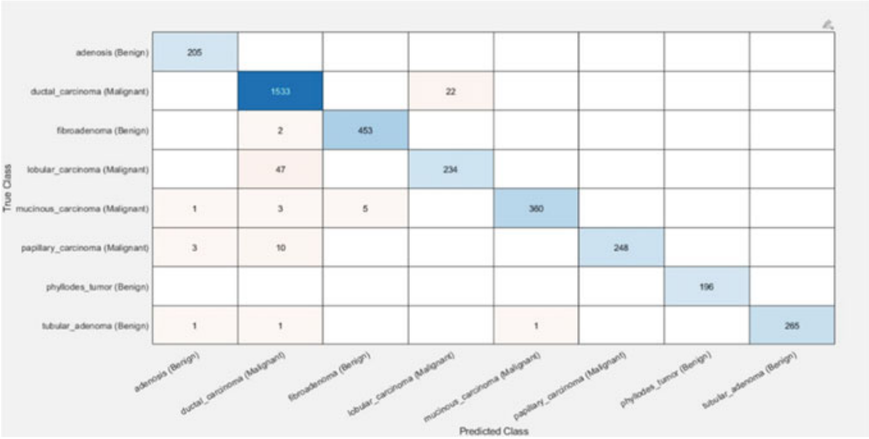


Fig. 7 CNN AlexNet confusion matrix for 30% validation histopathological data

Table 5 Using a histopathology database, SqueezeNet training accuracy, and training features were determined

Quantity	Value
Accuracy	93.73%
Elapsed time	124 m 20 s 2 h 4 m 20 s
Epochs	4
Iteration	348
Iteration per epoch	87
Validation frequency	87
Mini-batch size	96

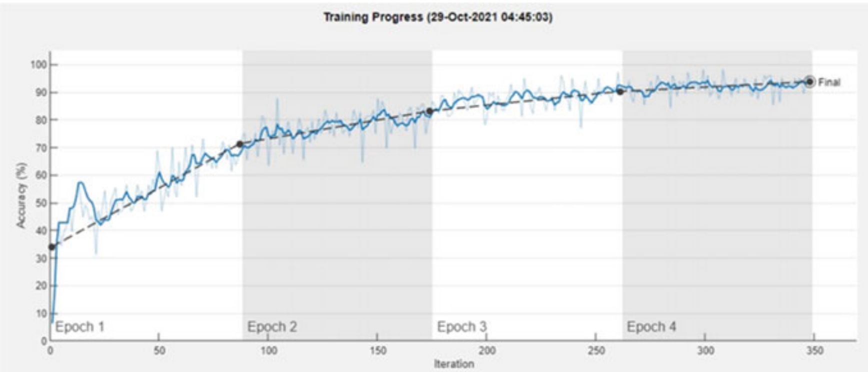


Fig. 8 Using 8 histopathological labeled data, the trained SqueezeNet model's accuracy

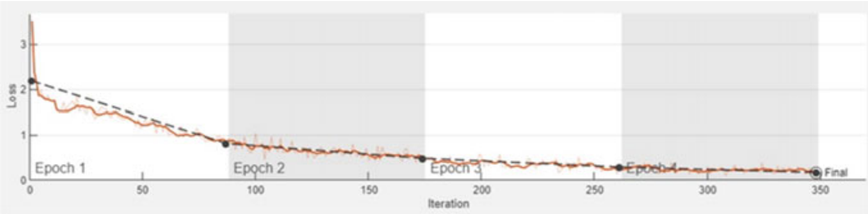


Fig. 9 For trained SqueezeNet with 8 labeled histopathological data

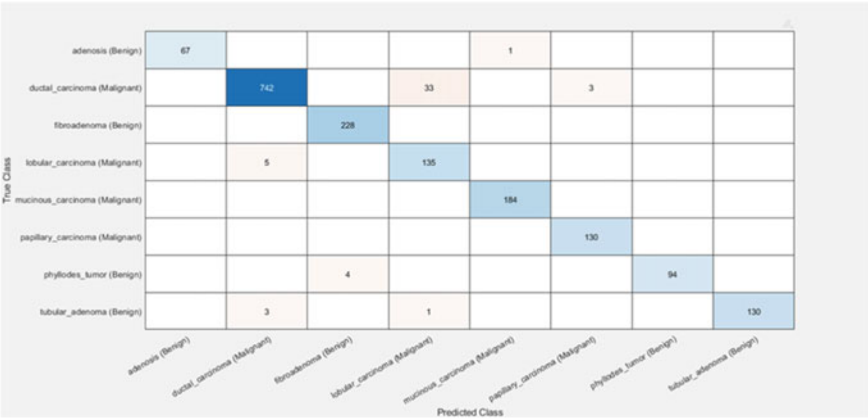


Fig. 10 CNN SqueezeNet confusion matrix for 30% validation histopathological data

3.3 CNN GoogLeNet Training Progress Labeled Histopathological Data

In this procedure, the data is trained using GoogLeNet, which is 22 layers deep, as well as an input layer with a size of 224 × 224 × 3, which contains the training results and features as shown in Table 6. The accuracy versus number of epochs is represented by the GoogLeNet training performance is shown in Fig. 11. The difference between the predicted value by the GoogLeNet CNN model, and the true value is termed as loss and is shown in Fig. 12. With histopathological data, the trained GoogLeNet confusion matrix is shown in Fig. 13.

Table 6 Using a histopathology database, GoogLeNet training accuracy, and training features were determined

Quantity	Value
Accuracy	96.18%
Elapsed time	209 m 23 s 3 h 29 m 23 s
Epochs	4
Iteration	348
Iteration per epoch	87
Validation frequency	87
Mini-batch size	96

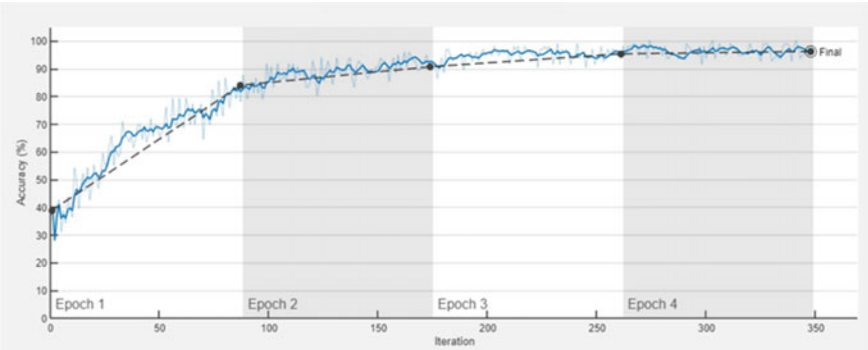


Fig. 11 Using 8 histopathological labeled data, the trained GoogLeNet model's accuracy

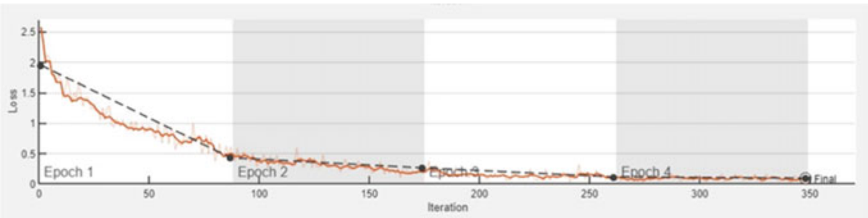


Fig. 12 For trained GoogLeNet with 8 labeled histopathological data, loss versus iteration

3.4 CNN ResNet18 Training Progress Labeled Histopathological Data

In this procedure, the data is trained using ResNet18, which is 18 layers deep, as well as an input layer with a size of 224 × 224 × 3, which contains the training results and features as shown in Table 7. The accuracy versus number of epochs is represented by the ResNet18 training performance as shown in Fig. 14. The difference between the predicted value by the ResNet18 CNN model, and the true value is termed as loss

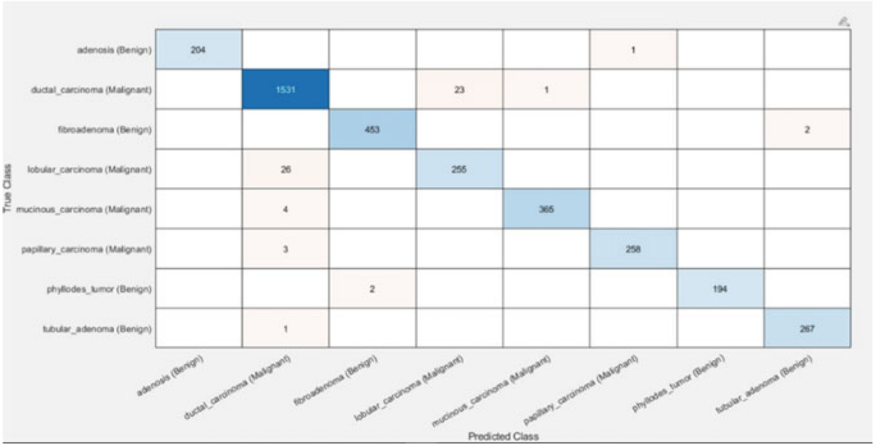


Fig. 13 CNN GoogLeNet confusion matrix for 30% validation histopathological data

Table 7 Using a histopathology database, ResNet18 training accuracy, and training features were determined

Quantity	Value
Accuracy	97.24%
Elapsed time	228 m 18 s 3 h 48 m 18 s
Epochs	4
Iteration	348
Iteration per epoch	87
Validation frequency	87
Mini-batch size	96

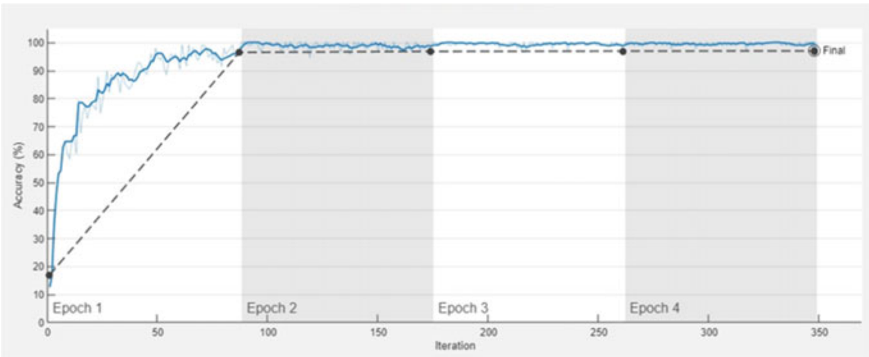


Fig. 14 Using 8 histopathological labeled data, the trained ResNet18 model's accuracy

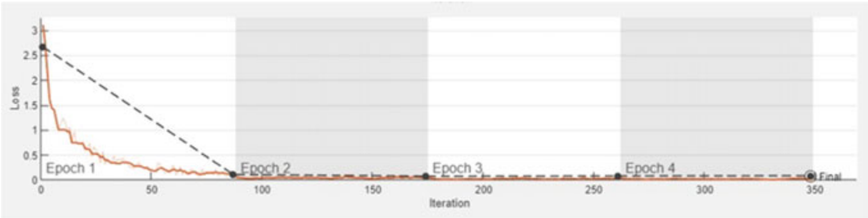


Fig. 15 For trained ResNet18 with 8 labeled histopathological data, loss versus iteration

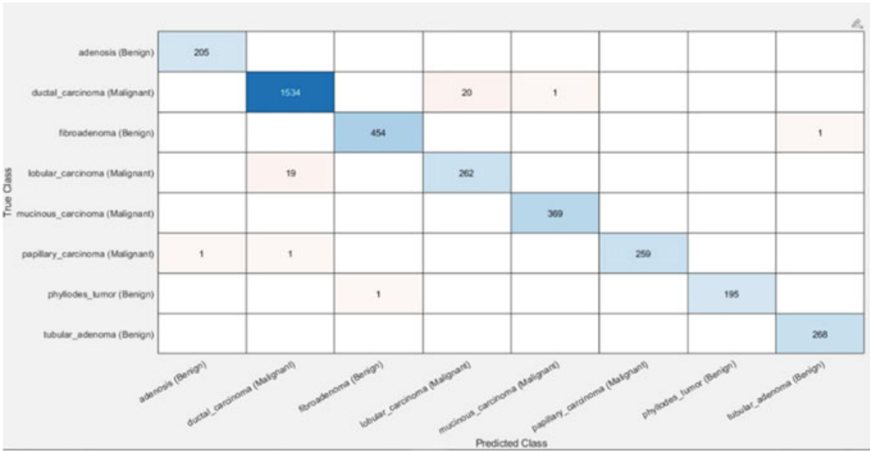


Fig. 16 CNN ResNet18 confusion matrix for 30% validation histopathological data

and is shown in Fig. 15. With histopathological data, the trained ResNet18 confusion matrix is shown in Fig. 16.

4 Result of Training Pre-trained AlexNet, SqueezeNet, GoogLeNet, and ResNet18 CNN Using Histopathological Data

The results of computing the confusion matrix with AlexNet, SqueezeNet, GoogLeNet, and ResNet18 CNN are given in Table 8. AlexNet for histopathological database gives a high accuracy of 99.33%, sensitivity of 96.61%, specificity of 99.49%, precision of 96.61%, error rate of 0.67%, and F-score of 96.61%. SqueezeNet gives a high accuracy of 99.29%, sensitivity of 97.91%, specificity of 99.57%, precision of 97.91%, error rate of 0.71%, and F-score of 97.91%. GoogLeNet gives a high accuracy of 99.56%, sensitivity of 98.08%, specificity of 99.68%, precision of 98.08%, error rate of 0.44%, and F-score of 98.08%. ResNet18 gives a

Table 8 Results of calculated CNN statistics for histopathology database of testing cases of study

CNN model	CNN measure					
	Accuracy	Recall or Sensitivity	Specificity	Precision	Error rate	F-score
SqueezeNet	0.9929	0.9791	0.9957	0.9791	0.0071	0.9791
AlexNet	0.9933	0.9661	0.9949	0.9661	0.0067	0.9661
GoogLeNet	0.9956	0.9808	0.9968	0.9808	0.0044	0.9808
ResNet18	0.9969	0.9880	0.9979	0.9880	0.0031	0.9880

high accuracy of 99.69%, sensitivity of 98.80%, specificity of 99.79%, precision of 99.80%, error rate of 0.31%, and F-score of 99.80%. From comparing the testing results of the four CNN architecture, the system with the proposed minimum batch size and maximum number of epochs results in high accuracy for all CNN architecture in the given application cases of study.

5 Conclusion

Breast tumor diagnosis using histopathology database the proposed CNN training method utilizing histopathology image data is one of the most effective technique used in breast tumor diagnosis. Different CNN architecture is used such as AlexNet, SqueezeNet, GoogLeNet, and ResNet18 are four pre-trained CNN architectures that have been utilized with 8 labeled data to diagnose breast cancer with great accuracy during training and testing. Data augmentation methods such as image cropping and adding noise to the given database to increase the number of training database images are introduced in the study, as given in Table 8. Hence, the two models can be used with histopathology data for diagnosis tumor in breast where both models give very high accuracy as shown in Table 8.

References

1. Spanhol FA, Oliveira LS, Petitjean C, Heutte L (2016) A dataset for breast cancer histopathological image classification. IEEE Trans Biomed Eng 63(7):1455–1462
2. Pathak R, Jha A, Neupane P, Chalise S, Basnyat A (2016) Histopathological evaluation of carcinoma of breast. J. Pathol. Nepal 6(11):922–927
3. Talei A, Akrami M, Mokhtari M, Tahmasebi S (2012) Surgical and clinical pathology of breast diseases. Histopathol Rev Recent Adv (3)
4. Definition of ductal carcinoma—NCI Dictionary of Cancer Terms—National Cancer Institute. [Online]. Available: https://www.cancer.gov/publications/dictionaries/cancer-terms/def/ductal-carcinoma. Accessed 15 Aug 2021
5. Lobular carcinoma in situ (LCIS)—Symptoms and causes—Mayo Clinic. [Online]. Available: https://www.mayoclinic.org/diseases-conditions/lobular-carcinoma-in-situ/symptoms-causes/syc-20374529. Accessed 16 Aug 2021

6. Dumitru A et al (2015) Mucinous breast cancer: a review study of 5 year experience from a hospital-based series of cases. Mædica 10(1):14
7. Martin P, Schlumberger J, Papillary thyroid carcinoma
8. Mishra SP, Tiwary SK, Mishra M, Khanna AK (2013) Phyllodes tumor of breast: a review article. ISRN Surg 2013:1–10
9. Zuhair A-R, Maron A-R (2014) Tubular adenoma of the breast: a case report. Case Rep Clin Med 03(06):323–326

A Vertical Handover Approach Using GTMA in Wireless Networks

Gaganpreet Kaur, Raman Kumar Goyal, and Rajesh Mehta

Abstract The smooth transfer of user services from an existing network to a new network in vertical handover for providing better quality of experience (QoE) to the users is a challenging task in the field of mobility management. To achieve that level of the QoE, the subscribers of heterogeneous networks may be forced to change the access network or the network operator. For better user experience, network parameters like throughput, packet loss rate (PLR), cost, jitter, and delay, etc., are considered in vertical handover decisions. In this paper, a graph theory and matrix approach (GTMA) based on a multi-attribute decision-making (MADM) mechanism is proposed for the ranking of the candidate networks to handle the issues in vertical handover. The numerical analysis of the proposed approach is performed using conversational traffic in heterogeneous networks. The proposed approach is compared with gray rational analysis (GRA), and the comparative results have revealed that proposed approach is superior to GRA.

Keywords GTMA · QoE · Vertical handover · Mobility management · MADM

1 Introduction

Wireless networks as well as mobile terminals are developing rapidly in order to connect to the best network for achieving good QoE with minimum delay, jitter, PLR, cost, and maximum throughput. Due to the rapid development in digital communication technology, mobility has been increased in the application scenarios, which leads to various issues such as handover failures and unnecessary handovers. To

G. Kaur (✉) · R. Kumar Goyal · R. Mehta
Thapar Institute of Engineering and Technology, Patiala, India
e-mail: gaganpreet.k@thapar.edu

R. Kumar Goyal
e-mail: ramankumar.goyal@thapar.edu

R. Mehta
e-mail: rajesh.mehta@thapar.edu

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_50

avoid the mobility issues and to ensure the better QoE for the users, the handover should be triggered at the appropriate time, and the network with the highest quality of service (QoS) should be selected for handover. Traditional handover techniques relied solely on received signal received power (RSRP) or received signal strength indicator (RSSI) for the decision of handover triggering and target network selection. Interference can easily disrupt this type of handover process, which results in unnecessary handovers. So, a handover decision should be made on the basis of multiple properties of the networks, which can be envisioned as a MADM problem [1–3]. MADM is a mathematical tool that can be utilized for decision-making on the basis of multiple attributes. Thus, the MADM can be used for selecting the most appropriate network from the available networks in relation to multiple attributes [2]. To handle and optimize the vertical handover problem, different kinds of MADM techniques [1–4] have been proposed by various researchers in the literature. Zhong et al. [1] presented a cross-layer framework for network selection on the basis of cognitive cycle and a MADM method (combination of analytic hierarchy process (AHP) and Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS)). This method has reduced the number of handovers and ranking abnormality rate as compared to traditional methods. Liu et al. [2] described a QoS-aware handover approach that incorporates the benefits of both TOPSIS as well as fuzzy logic. The fuzzy logic is utilized to diminish the effect of uncertain weight values. The algorithm triggers the handover on the basis of the closeness coefficient of the TOPSIS. Mansouri et al. [3] described a handover method which integrates fuzzy logic and TOPSIS. In this method, FANP is utilized for assigning weights to the network attributes. The authors have compared the performance with the traditional handover methods. Labhy et al. [4] presented a hybrid handover technique which is combination of the TOPSIS method and utility function. The performance of each alternative network according to the different network traffic classes is calculated using TOPSIS method. The utility function is utilized for selecting the best access network by taking into account the preferences of users according to the type of service they are using. Cicioglu [5] has proposed entropy-based simple additive weighting decision-making method for selection of small cells in 5G. This is a centralized and proactive technique. The decisions are taken by centralized controller. This method selects the most appropriate cell and allocates it to relevant the mobile nodes on the basis of three factors such as bandwidth, signal interference to noise ratio, and user density. Subramani et al. [6] have introduced a two-stage fuzzy logic-based vertical handover technique. Stage 1 involves performing a fuzzy logic-based handover requirement analysis based on data rate, received signal strength, and latency. This stage reduces the unnecessary handovers. In stage 2, the target access network is ranked and selected using fuzzy-AHP and fuzzy-TOPSIS. Alhabo et al. [7] have proposed GRA-based handover technique. In this paper, AHP is utilized for obtaining weights of the handover metrics. The main contributions of the paper are listed below:

- MADM algorithm, referred as GTMA, is proposed for ranking the candidate networks.

- The GTMA is applied using numerical simulation for ranking the wireless networks in heterogeneous environment for conversational traffic.
- The proposed technique is compared with GRA based on ranking abnormality.

The rest of the paper is organized as: the details of the proposed GTMA ranking technique are provided in Sect. 2. The proposed approach is compared with GRA [7] in Sect. 3. Finally, the conclusion of the work presented as well as research directions are described.

2 Proposed Approach

Graph theory is systematical and logical representation applied in numerous types of systems for modeling and analyzing [8]. The matrix approach is beneficial for analyzing the graph models in order to derive the system function and index needed to achieve the objectives [8, 9]. The detailed flow chart of the proposed approach is shown in Fig. 1.

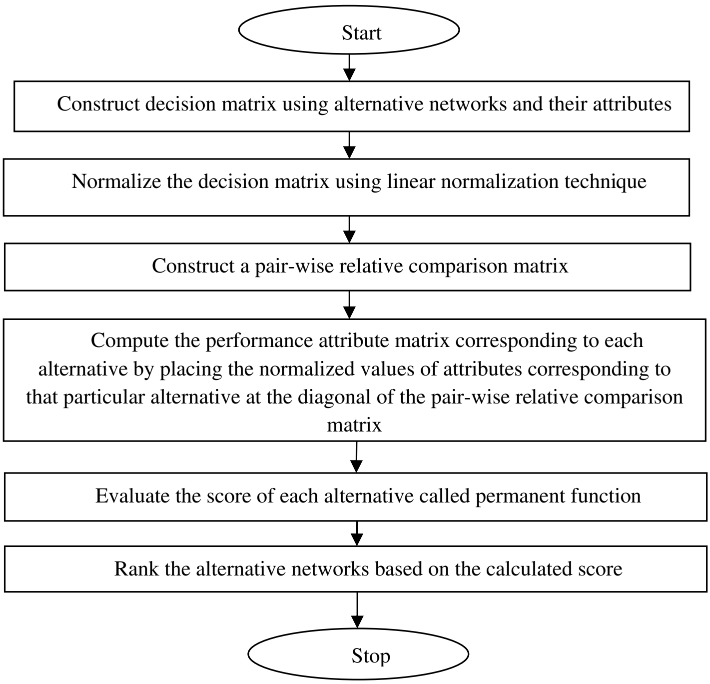


Fig. 1 Flow chart of the proposed approach

Table 1 Range of the network parameters corresponding to each network

	Delay (ms)	Jitter (ms)	Throughput (Gbps)	Price (1–10)	PLR (%)
UMTS	20–40	5–10	0.0001- 0.002	1–6	0.0025–0.009
LTE-A	10–30	3–10	0.1–0.3	4–7	0.0010–0.0015
5G	1–20	1–8	0.1–1	8–10	0.0010–0.0014
WLAN	100–150	10–20	0.001–0.011	1–6	0.0020–0.008

Table 2 Decision matrix between the available networks and their attributes

	UMTS	LTE-A	5G	WLAN
Delay (ms)	25.71678	11.51709	7.598124	108.1091
Jitter (ms)	8.786001	3.377651	4.981766	17.94285
Throughput (Gbps)	0.001532	0.20616	0.522452	0.004112
Price (1–10)	2.902229	5.558334	8.023804	3.642666
PLR (%)	0.006191	0.001467	0.001135	0.002994

The numerical analysis of proposed approach is performed using a heterogeneous wireless network which includes wireless local area network (WLAN), long-term evolution-advanced (LTE-A), Universal Mobile Telecommunications System (UMTS), and fifth generation (5G) networks. Initially, the mobile terminal was connected to LTE-A network. In this paper, PLR, throughput, jitter, price, and delay are considered as handover decision criteria. A case of conversational traffic is considered in the numerical analysis [1]. The range of attribute values corresponding to each network is provided in Table 1 [10–15].The stepwise explanation of the proposed approach is as:

Step 1: Construction of Decision Matrix

Mobile terminal generates a decision matrix between the available networks and their parameters at a given decision point is shown in Table 2.

Step 2: Normalization of the Decision Matrix

The different network attributes have different dimensions. This arises the need of normalizing the decision matrix in such a way that all the attributes will have values in the uniform range. By considering the impact of the different attributes on the target network, the attributes are categorized as: favorable attributes and non-favorable attributes.

The favorable attributes are those for which the higher values are preferred, examples of such attributes are throughput, bandwidth, signal strength, etc. To normalize the decision matrix of Step1, linear normalization technique is used. The favorable and non-favorable attributes are normalized using Eqs. (1) and (2), respectively, [1, 3, 4, 8, 9].

Table 3 Normalized decision matrix corresponding to Table 2

	UMTS	LTE-A	5G	WLAN
Delay (ms)	0.295454	0.659726	1	0.070282
Jitter (ms)	0.384436	1	0.678003	0.18825
Throughput (ms)	0.002932	0.3946	1	0.007871
Price (ms)	1	0.52214	0.361702	0.796732
PLR (%)	0.183311	0.773582	1	0.379055

$$nd_{\text{favorable}} = \frac{d_{ij}}{\max_i (d_{ij})} \tag{1}$$

$$nd_{\text{non - favorable}} = \frac{\min_i (d_{ij})}{d_{ij}} \tag{2}$$

Here, $nd_{\text{favorable}}$ corresponds to the normalized values of the favorable attributes, and $nd_{\text{non - favorable}}$ represents the normalized values of non-favorable attributes.

Here, the attributes delay, price, jitter, and PLR are non-favorable attributes, whereas throughput is a favorable attribute. The decision matrix after normalization is tabulated in Table 3.

Step 3: Construction of Pair-wise Comparison Matrix

The pair-wise comparison matrix is constructed between attributes of the networks. For the construction of pair-wise comparison matrix, following steps are followed:

I. **Graphical representation**

The interrelationship of attributes is represented by graph. Each node in graph represents the attribute considered for network selection and their relative importance is represented by the edges between them. The number of nodes in the graph is equal to the number of network selection attributes taken into consideration. A directed edge from node 'u' to node 'v' (ed_{uv}) denotes the relative importance of a node u over another node v in the network selection. If there are four attributes (N_1, N_2, N_3, and N_4) and all of these are interdependent, then their graphical representation is shown in Fig. 2 [8, 9].

II. **Matrix representation**

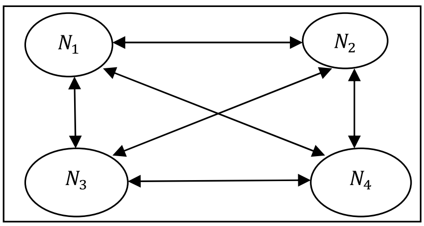


Fig. 2 Graphical representation of attributes and their interrelationship

Matrix is a way to represent the one-to-one relationship among the network selection attributes shown in Fig. 2. If p_{uv} denotes the relative importance, uth attribute is having over the vth attribute, then Eq. (3) is used to evaluate the relative importance, and vth attribute is having over the uth attribute.

$$p_{vu} = 1 - p_{uv} \tag{3}$$

The relative importance among the two attributes (p_{uv}) is given values on the scale of 0–1 and is shown in Table 4. The relative importance among the network attributes depends upon type of network application. The conversational traffic is more concerned with delay and jitter. The preferences of the users and price and user preferences are also considered. The pair-wise comparison matrix of the network attributes for conversational traffic [12, 13] is as shown in Table 5.

Step 4: Computing Performance Attribute Matrix (PAM)

The performance attribute matrix is computed for each alternative by placing the normalized values of the attributes corresponding to that specific alternative at the diagonal of the relative comparison matrix. The performance attribute matrix corresponding to each alternative for conversational type network traffic is shown in Tables 6, 7, 8, and 9.

Table 4 Relative importance of the decision attributes [8, 9]

Class description	p_{uv}	$p_{vu} = 1 - p_{uv}$
Two attributes are equally important	0.500	0.5
One attribute is slightly more important than other	0.590	0.410
One attribute is more important over than other	0.665	0.335
One attribute is highly important than other	0.745	0.255
One attribute is extremely more important than other	0.865	0.135
For median values	0.545, 0.627, 0.705, 0.805	0.455, 0.373, 0.295, 0.195

Table 5 Pair-wise comparison matrix for conversational traffic

	Delay	Jitter	Throughput	Price	PLR
Delay	-	0.5	0.59	0.627	0.745
Jitter	0.5	-	0.545	0.59	0.705
Throughput	0.41	0.455	-	0.373	0.545
Price	0.373	0.41	0.627	-	0.665
PLR	0.255	0.295	0.455	0.335	-

Table 6 Performance attribute matrix for UMTS

	Delay	Jitter	Throughput	Price	PLR
Delay	0.295454	0.5	0.59	0.627	0.745
Jitter	0.5	0.384436	0.545	0.59	0.705
Throughput	0.41	0.455	0.002932	0.373	0.545
Price	0.373	0.41	0.627	1	0.665
PLR	0.255	0.295	0.455	0.335	0.183311

Table 7 Performance attribute matrix for LTE-A

	Delay	Jitter	Throughput	Price	PLR
Delay	0.659726	0.5	0.59	0.627	0.745
Jitter	0.5	1	0.545	0.59	0.705
Throughput	0.41	0.455	0.3946	0.373	0.545
Price	0.373	0.41	0.627	0.52214	0.665
PLR	0.255	0.295	0.455	0.335	0.773582

Table 8 Performance attribute matrix for 5G

	Delay	Jitter	Throughput	Price	PLR
Delay	1	0.5	0.59	0.627	0.745
Jitter	0.5	0.678003	0.545	0.59	0.705
Throughput	0.41	0.455	1	0.373	0.545
Price	0.373	0.41	0.627	0.361702	0.665
PLR	0.255	0.295	0.455	0.335	1

Table 9 Performance attribute matrix for WLAN

	Delay	Jitter	Throughput	Price	PLR
Delay	0.070282	0.5	0.59	0.627	0.745
Jitter	0.5	0.188245	0.545	0.59	0.705
Throughput	0.41	0.455	0.007871	0.373	0.545
Price	0.373	0.41	0.627	0.796732	0.665
PLR	0.255	0.295	0.455	0.335	0.379055

Step 5: Evaluate the Permanent Function

The score of each alternative also called as permanent function is computed similarly as determinant of matrix is calculated, provided all minus (−) signs are replaced with plus (+) signs in the determinant equation [5, 6]. The permanent function of Tables 6, 7, 8, and 9 is presented in Table 10.

Table 10 Permanent functions and ranking of candidate networks

Candidate networks	Permanent function	Rank
UMTS	2.818192	3
LTE-A	3.76233	2
5G	4.083524	1
WLAN	1.5119	4

Step 6: Ranking of the Alternatives

The ranking of alternatives is done using their permanent function. The alternatives are listed in the descending order of their permanent function. The alternative at the top of the list, i.e., the alternative with the maximum value of permanent function is given rank one, as this alternative is considered as the best option. Similarly, the alternative with least value of permanent function is given the last rank and considered as the worst option. It is observed from Table 10, 5G is the best alternative (rank 1) and WLAN is the worst alternative (rank 4) in the above scenario according to ranking provided by GTMA (proposed approach).

3 Comparison of Proposed Approach (GTMA) with GRA

In this section, for comparison of proposed approach, GRA is applied on the above presented scenario, and the ranking obtained by GRA [7] and GTMA is presented in Table 11.

Ranking of candidate networks and ranking abnormality is two factors which are considered for comparison purpose. According to the ranking given by proposed approach 5G is assigned rank 1, whereas GRA has given rank 1 to LTE-A. Ranking abnormality is the state when the ranking of the available networks is changed after eliminating the lowest ranked network [1]. If WLAN network is removed, the ranking provided by proposed approach and GRA is presented in Table 11 and Fig. 3. It is inferred from Fig. 3 that after removal of WLAN network, the ranks of the available networks get changed in case of GRA, where as in case of proposed approach, the ranks of the available networks remain stable. Thus, GRA suffers from ranking

Table 11 Ranking provided by proposed approach (GTMA) and GRA

	Ranking by GTMA		Ranking by GRA	
	With WLAN	Without WLAN	With WLAN	Without WLAN
UMTS	3	3	3	3
LTE-A	2	2	1	2
5G	1	1	2	1
WLAN	4		4	

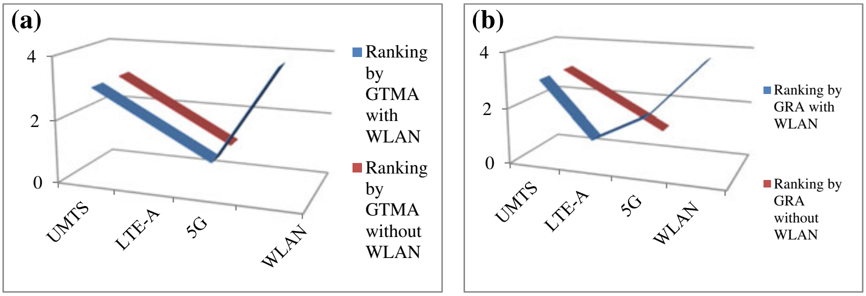


Fig. 3 Ranking with WLAN and without WLAN by **a** proposed approach and **b** GRA

abnormality, whereas the proposed approach has overcome the issue of ranking abnormality.

4 Conclusion

In this paper, a GTMA method is utilized for the ranking of the candidate networks. This proposed approach will reduce the number of unnecessary handovers; enhance the handover stability, user experience, utilization of resources, and QoS. This technique will provide ubiquitous connectivity which enables the users to stay connected to the Internet even when they are moving from one place to another. During their movement, the switching among wireless technologies will be seamless, i.e., the users will not experience any kind of disruption in their call or data session when it will be transferred from one base station to another. The numerical analysis is performed using heterogeneous wireless network consisting of WLAN, LTE-A, UMTS, and 5G networks. The comparative results revealed that the proposed approach outperforms the GRA by selecting the appropriate network along with reduction in the ranking abnormality. A single decision point analysis of proposed approach is performed in this paper. In the future, more decision points will be considered for detailed analysis of GTMA, and number of handovers will be computed along with issues of handover triggering.

References

1. Zhong Y, Wang H, Lv H (2020) A cognitive wireless access networks selection algorithm based on MADM. Ad Hoc Netw 109:1–8
2. Liu Q, Kwong CF, Zhang S et al (2018) A hybrid fuzzy-MADM based decision-making scheme for QoS aware handover. In: IET Conference proceedings. The Institution of Engineering and Technology, pp 1–5

3. Mansouri M, Leghris C (2020) A use of fuzzy TOPSIS to improve the network selection in wirseless multiaccess environments. J Comput Netw Commun
4. Lahby M, Sekkaki A (2017) Optimal vertical handover based on TOPSIS algorithm and utility function in heterogeneous wireless networks. In: 2017 international symposium on networks, computers and communications (ISNCC), pp 1–6, IEEE
5. Cicioğlu M (2021) Multi-criteria handover management using entropy-based SAW method for SDN-based 5G small cells. Wireless Netw 27(4):2947–2959
6. Subramani M, Kumaravelu VB, Murugadass A (2021) Fuzzy logic-based handover requirement analysis and access network selection for device-to-device communication. J Circuits Syst Comput 30(1)
7. Alhabo M, Zhang L, Nawaz N (2019) GRA-based handover for dense small cells heterogeneous networks. IET Commun 13(13):1928–1935
8. Geetha NK, Sekar P (2017) Graph theory matrix approach—a qualitative decision making tool. Mater Today Proc 4(8):7741–7749
9. Venkata Rao R (2006) A material selection model using graph theory and matrix approach. Mater Sci Eng A 431:248–255
10. Chattate I et al (2018) A fuzzy-AHP based approach for enhancing network selection in heterogeneous networks using battery energy criterion. Int J Eng Technol 7(4):118–123
11. Rahman MT, Chowdhury MZ, Jang YM (2016) Radio access network selection mechanism based on hierarchical modelling and game theory. In: 2016 international conference on information and communication technology convergence (ICTC), pp 126–131. IEEE
12. Lina Z, Qi Z (2014) Multiple attribute network selection algorithm based on AHP and synergetic theory for heterogeneous wireless networks. J Electron 31(1):29–40
13. Zheng S, Qi Z (2012) Network selection based on multiple attribute decision making and group decision making for heterogeneous wireless networks. J China Univ Posts Telecommun 19(5):92–114
14. Lahby M, Cherkaoui L, Adib A (2013) An enhanced-TOPSIS based network selection technique for next generation wireless networks. In: ICT 2013, pp 1–5. IEEE
15. Mouâd M, Cherkaoui L (2017) The use of MADM methods in the vertical handover decision making context. In 2017 international conference on wireless networks and mobile communications (WINCOM), pp 1–6, IEEE

Copy-Move Forgery Detection Using K-Means and Hu's Invariant Moments

N. B. S. P. S. Harshith, D. Sindhuja, Ch. Raghava Reddy, A. Deepthi, and G. GopaKumar

Abstract Image forensics is one of the most active research domains. As technology is advancing, we can add or take out crucial features from a picture without any trail of tampering. Therefore, its authenticity is called into question especially when images have impressive power. Copy-move is a kind of forgery, where portions of a picture are transformed and inserted into the same picture. Copy-move forgery detection is one such research domain that has put forward various methods to find out copy-move forgery. Many techniques based on image processing and machine learning have been put forward to detect the forgery. Since the duplicated parts are from the same image, many of the features will be similar to the rest of the image making it difficult to detect forgery using the latest methods. In this work, we propose to use SIFT keypoint-based forgery detection with clustering for quickly identifying copy-move forgeries in highly textured regions. As the SIFT keypoints are difficult to detect in smooth regions, we propose to use Hu's invariant-based block-based forgery detection strategy to detect the missing cases. We show that the joint approach outperforms the method reported by Li et al. (IEEE Trans Inf Forensics Secur 10:507–518, 2015) on the popular copy-move forgery detection dataset MICC-600.

N. B. S. P. S. Harshith (✉) · D. Sindhuja · Ch. Raghava Reddy · A. Deepthi · G. GopaKumar
Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, Kerala, India
e-mail: namburuharshith@am.students.amrita.edu

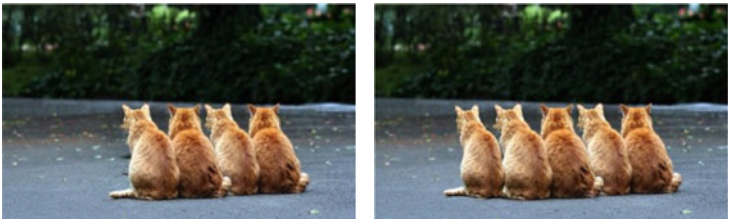
D. Sindhuja
e-mail: sindhujadantuluri@am.students.amrita.edu

Ch. Raghava Reddy
e-mail: raghava@am.students.amrita.edu

A. Deepthi
e-mail: akulasrideepthi@am.students.amrita.edu

G. GopaKumar
e-mail: gopakumarg@am.amrita.edu

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_51



(a) Original cats image (b) Forged cats image

Fig. 1 Cats image before and after forgery

1 Introduction

Images are how we see the world today. Anything we do is captured through images. They are more than what capture moments are in our lives. They are used in investigations as pieces of evidence and used professionally to showcase photographic talent. But as the technology grows and the Internet advances, the ability to get access to editing software like Photoshop or online tools becomes easy. This has resulted in easier ways for image tampering. These alterations can be used to beautify and increase the visual effects but are also used to tamper the evidence in court, media, etc., which cannot be detected with the naked eye. These days, image tampering is widely seen as a threat to the authenticity of images. To reduce this evidence tampering, image forensics is taking appropriate measures to detect forged images in an efficient way.

The most common categories of image forgeries are splicing and copy-move forgeries. Copy-move is a special method, where a part or segment of an image is copied and inserted into the same image in a different area. By this, we can conceal or duplicate part of an image leading to image tampering.

As shown in Fig. 1, the copied region shares the characteristics like the color, texture and noise of the original image, it is hard to detect the forged part(s). Tampering with evidence and hiding facts rapidly increase for spreading false news. Traditional methods to detect copy-move forgery are categorized into keypoint and block-based methods. While block-based methods result in high accuracy, they are computationally expensive. Keypoint-based methods have less time complexity but often compromise the accuracy. Taking these into consideration, we put forward a combined method for copy-move forgery detection.

2 Related Works

Within the past few years, different methods were developed to detect a forgery in images. There are many ways to forge an image but we focus on copy-move forgery. The traditional non-deep learning-based image forgery detection methods are broadly grouped into the block and keypoint-based methods. The keypoint-based methods identify keypoints and derive features at these keypoints for detecting forgery. In block-based methods, arbitrary/fixed shaped/sized blocks are extracted, and their features are utilized to identify forgery.

Keypoint-based methods identify copy-move forgery by keypoint comparisons. As exhaustive pixel or block-based comparison is not happening in keypoint-based, these methods are less expensive than block-based methods. Fast copy-move forgery detection using scale-invariant feature transform (SIFT [1]) keypoints was put forward by Gouda [2]. SIFT features are scale, rotation and brightness invariant and clustered using fuzzy c-means (FCM). Li et al. [3] proposed a method to detect forgery using segmentation with CMFD and keypoints for extracting features from the segments and further used refined transform matrix via EM-based algorithm to eliminate false alarm patches.

In block-based methods, there are different ways to detect image forgery in which an image is split into overlapping or non-overlapping blocks. Bashar and Noda [4] proposed an algorithm to detect duplicated regions in natural images using block-based methods. Here, wavelet transform and KPCA [4] with RBF kernel is used to detect features per block of the image. Jan and Tariq [5] put forward a method that uses K-means clustering and hybrid transform. DWT reduces image size and DCT diminishes feature dimensions. Blocks are grouped using K-means clustering, and feature matching is done to detect forgery.

As noted earlier, using keypoints detection is fast and robust to different transformations of copy-moved areas. However, these methods offer less accuracy when compared to the computationally intensive block-based methods. We put forward a hybrid algorithm where keypoint-based detection is employed to detect easy cases and use block-based methods to ensure that the image is free from forgery. We propose to use color-based clustering [6] with SIFT [2]-based keypoints to identify the copy-move forgery. These are very good feature points that are invariant to rotation, illumination and scaling. Though the SIFT-based forgery detection could quickly identify copy-move forgery in many cases, the method fails to identify for regions such as smooth regions, where it is hard to detect keypoints. To tackle such cases, we need a block-based feature set. Hu's invariant moments is one of the best feature sets for object comparison and is invariant to a set of geometric transformations including translation, rotation and scaling. We propose to use these feature sets for identifying copy-move forgeries in the next level for the cases where SIFT-based forgery detection failed. We show that the proposed combined approach outperforms the result reported by Li et al. [3] on the popular copy-move forgery dataset MICC-600 [7] on a number of metrics including precision, F-score and accuracy Table 4.

3 Proposed Solution

In this section, we discussed the suggested framework in depth. The main steps of the proposed method are clustering, keypoint extraction, matching and forgery detection.

Figure 2 gives an overview of the framework, where SIFT keypoint-based detection on clusters identified by K-means is used to identify the easy cases of forgery. If failed, the image is processed using the block-based method with Hu's invariant moments to ensure that the image is really free from forgery. The following subsections explain this process-flow in detail.

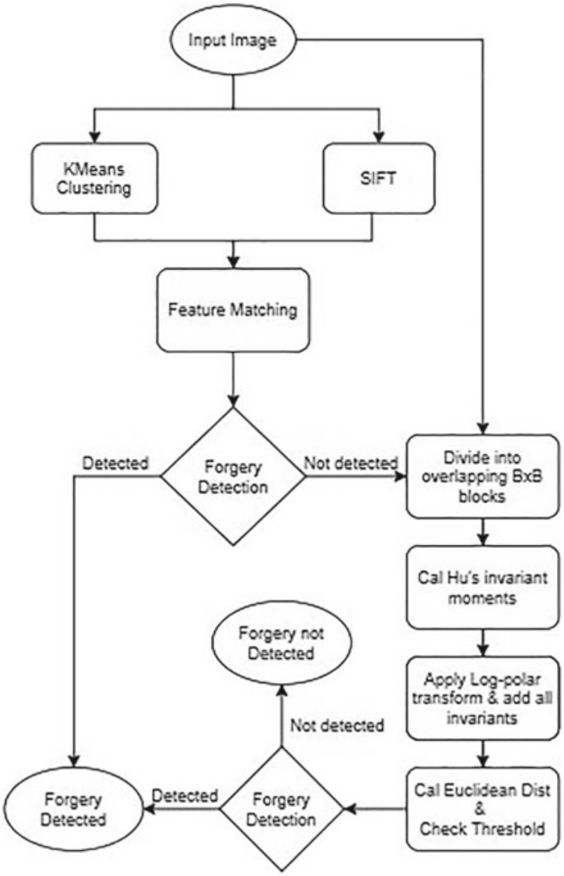


Fig. 2 Flowchart for the proposed algorithm

3.1 Keypoint-Based Forgery Detection for Easy Cases

The input image is clustered based on the color feature using K-means [6] clustering. This will cause similar color sections of both the copy-moved region and its source region to fall in the same cluster. We compare the SIFT keypoints with every other keypoints falling in the same cluster and look for potential matches. Given the capacity of SIFT to uniquely identify a keypoint [2], a good number of matched keypoints indicates that an image is forged. If the image is not identified as a forged one, it is passed to the block-based forgery detection phase.

3.2 Block-Based Forgery Detection for Difficult Cases

As discussed earlier, if the keypoint detection-based method identifies that the image is authentic, a block-based method is then employed to double check this. The image is split into 4×4 overlapping blocks and each block is compared with every other block using Hu's invariant moments [8]. We calculate the Hu's moments up to seventh order. These features are invariant to rotation, scaling and translation. The seventh Hu's moment just shifts the sign for the reflection transformation. Hence these are excellent features to identify forgery under different geometric transformations.

4 Result and Analysis

In this section, we discuss the results obtained by the proposed algorithm. We test the efficacy of the keypoint-based track and the combined approach using the MICC-600 dataset [7]. The results are then compared with the results reported in [3] and show that the suggested approach outperforms the procedure by Li et al. [3] on a number of metrics including precision, accuracy, false positive rate (FPR) and false negative rate (FNR).

The dataset used in this study is the popular copy-move forgery detection dataset MICC-600 [7]. This dataset has sizes of the images varying between 800×532 and 3888×2592 pixels and the size of the tampered region also vary across images. There are 440 original images and 160 forged images in the dataset.

4.1 Performance of Keypoint Based Algorithm

Initially, the images in the dataset are sent to the keypoint-based detection module for detection. Here, the images are clustered based on the color information of the pixels. The number of clusters is decided as seven using the Elbow method [1].

Table 1 Confusion matrix of keypoint-based results

Confusion matrix	Actual (forged)	Actual (original)
Detected (forged)	121	47
Detected (original)	39	393

Table 2 Results obtained using keypoint-based algorithm

	Accuracy	Precision	Recall
Keypoint-based track (%)	85.66	72.02	75.62

The clustering helps to fall the same colored regions in the same clusters. We extract SIFT keypoints in parallel and SIFT descriptors for each region falling in the same clusters are compared. The primary check for forgery detection is done by using SIFT keypoints. The Euclidean distance measure is used to match SIFT keypoints. The regions are identified as forged if the keypoints matched between the blocks are more than the adaptive threshold calculated as mentioned in [9].

Table 1 summarizes the detection accuracy of the keypoint-based detection track. The accuracy, precision and recall of the keypoint-based algorithm are recorded in Table 2. There are 39 images wrongly identified as pristine, while 47 are wrongly identified as forged. An example image where the keypoint-based track could easily detect the forgery is given in Fig. 3. As expected, note that many keypoints are matched for the copy-moved regions. However, the keypoint-based track failed for the image shown in Fig. 4 as the forgery was introduced to a less textured region,

Fig. 3 Example image for successful forgery detection by keypoint-based track

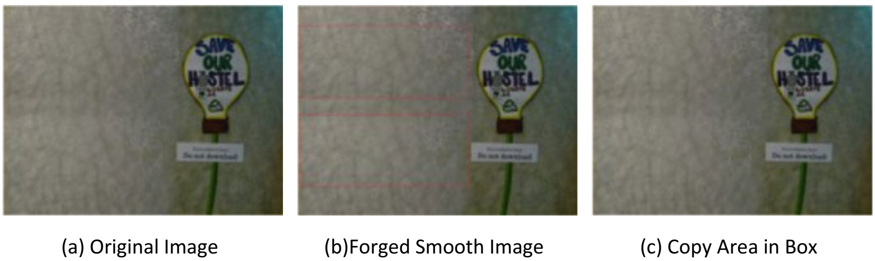


Fig. 4 Example image for failed forgery detection by keypoint-based track

Table 3 Confusion matrix of obtained results

Confusion matrix	Actual (forged)	Actual (original)
Detected (forged)	143	51
Detected (original)	17	389

Table 4 Comparison of results between paper [4] and our paper

	Accuracy	Precision	Recall	FNR	FPR
Li et al. [4] (%)	86.60	69.80	88.10	11.90	13.80
Proposed Alg (%)	88.67	73.71	89.37	19.30	10.45

where we were hardly able to detect sufficient SIFT keypoints for comparison.

4.2 Performance of Block-Based Algorithm

As discussed in earlier sections, if the image is identified as pristine in the keypoint-based forgery detection phase, it is passed through the block-based strategy [10–12] to double-check whether it is free from forgery. We divide the image into 4×4 blocks, and Hu's invariant moments are extracted for each block. The log-polar transform of the features is computed, summed up and compared using the Euclidean distance to decide the block has undergone copy-move forgery. The block size (4×4) is empirically found, and the threshold on Hu's moments is taken from [8].

After testing with the proposed algorithm, we found that 389 original images were properly identified as unforged, and 51 images were identified to be forged. Out of the 160 forged images, we detected 143 as forged images and 17 as original images as shown in Table 3 giving an accuracy of 88.67%.

We have also compared the result with the results reported in [3]. They segmented the image into semantic-independent patches. Thus, the copy-move forgery can be resolved by partially matching those patches. It can be seen from Table 4 that our proposed strategy outperformed [3] in several evaluation metrics such as precision and recall. The reason why we have a higher accuracy is that we are following a two-layer forgery detection method in which we split an image into 128 blocks of size 4×4 and matching blocks. Hence, small forged regions were also being detected through Hu's invariant moment with fewer false positives. However, if the copy-moved area is tiny, the method may fail. One image for which the method failed is shown in Fig. 5.

(a) Original Image (b) Copy Area in Box (c) Forged Image

Fig. 5 Forgery image not identified by our algorithm

5 Conclusion

This paper has explored a hybrid approach to detect copy-move forgery that used both keypoint and block-based methods [13–15]. The proposed approach used clustering and SIFT keypoints-based forgery detection followed by Hu's invariant-based block comparison. The SIFT keypoints-based comparison could quickly identify forgery especially in highly textured regions, where the Hu's moments could easily work for smooth regions. Also, we have employed clustering which helped to reduce the number of comparisons required in the keypoint-based track. We showed that the proposed method outperformed the method in [3] on the MICC-600 dataset (Table 1). Further, this method can be improved by using a 2X2 block in block-based method.

References

1. Guo F, Yang J, Chen Y, Yao B (2018) Research on image detection and matching based on SIFT features. In: 2018 3rd International conference on control and robotics engineering (ICCRE)
2. Hesham AA, Hegazy AA, Gouda IS (2018) A fast SIFT based method for copy move forgery detection. Future Comput Inform J 3(2)
3. Li J, Li X, Yang B, Sun X (2015) Segmentation based image copy move forgery detection scheme. IEEE Trans Inf Forensics Secur 10(3):507–518. https://doi.org/10.1109/tifs.2014.2381872
4. Bashar M, Noda K, Ohnishi N, Mori K (2010) Exploring duplicated regions in natural images. IEEE Trans Image Process. doi:https://doi.org/10.1109/TIP.2010.2046599
5. Jan Shah T, Tariq Banday M (2017) Copy-move forgery detection using hybrid transform and K means clustering technique. In: 2017 3rd International conference on applied and theoretical computing and communication technology (iCATccT). doi: https://doi.org/10.1109/ICATCCT.2017.8389110
6. Chitade A, Kumar Katiyar S (2010) Color based image segmentation using K-means clustering. In: 2010 International journal of engineering science and technology
7. Amerini I, Ballan L, Caldelli R, Del Bimbo A, Serra G (2011) A SIFT based forensic method for copy-move attack detection and transformation recovery. IEEE Trans Inf Forensics Secur 6(3):1099–1110
8. Tejas K, Swathi C, Rajesh Kumar M (2018) Copy move Forgery using Hu's invariant moments and log-polar transformations. In: 2018 3rd IEEE International conference on recent trends in electronics, information communication technology (RTEICT)

9. Pun C, Yuan X, Bi X (2015) Image forgery detection using adaptive oversegmentation and feature point matching. IEEE Trans Inform Forensics Secur 10(8):1705–1716
10. Shivsubramani K, Loganathan R, Srinivasan CJ, Ajay V, SomanKP (2007) Multiclass hierarchical SVM for recognition of printed tamil characters
11. Subathra P, Baskar A, Kumar DS (2012) Detecting digital image forgeries using resampling by automatic region of interest (ROI). CTACT J Image Video Process 2(4)
12. Lekha HPS, Nair SMA, Unni R, Priya HV, Poornachandran P (2013) Digital image forgery detection on artificially blurred images. In: International conference on emerging trends in communication, control, signal processing computing applications (C2SPCA), Bangalore, India
13. Donna AA, Shilpa MA, Gaina KGb (2015) Hybrid approach for anomaly detection. Int J Appl Eng Res 10:2170–2174
14. Menon SS, Mary Saana NJ, Deepa G (2019) Image forgery detection using hash functions. Int J Rec Technol Eng (IJRTE) 8(1)
15. Narayanan SS, Gopakumar G (2020) Recursive block based keypoint matching for copy move image forgery detection. In: 2020 11th International conference on computing, communication and networking technologies (ICCCNT), 2020, pp 1–6. doi: https://doi.org/10.1109/ICCCNT49239.2020.9225658

Urban Sound Classification Using Adaboost

Anam Bansal and Naresh Kumar Garg

Abstract Classifying environmental sounds such as gunshots and dog barking are gaining popularity. Environmental sound classification(ESC) helps in developing context-aware applications such as security systems and criminal investigation systems. Research in speech and music has been done but environmental sounds are different because of their unstructured nature and attracts extensive attention in the field of research. Researchers have explored various preprocessing techniques, feature extraction and feature selection methods, and classification algorithms for ESC. In this paper, the ensemble technique—Adaboost algorithm— is applied to classify environmental sounds. The accuracy of different base estimators is evaluated on the publicly available dataset UrbanSound8K, and the highest accuracy is obtained in the case of the base estimator as random forest. The results of the Adaboost algorithm are also compared with the benchmark results reported using other machine learning classification algorithms such as support vector machines(SVM), IBK5, random forest 500, J48, and ZeroR.

1 Introduction

One of the challenging issues in context-aware systems is to detect the type of sounds produced in the environment. ESC is the trending area of research. Audios have become prevalent and are preferred over videos and text. Audios have helped in several applications in everyday lives such as gunshot detection, wildlife monitoring, and architectural design of concert halls. Audio scene recognition(ASR) and audio event recognition(AER) are upcoming areas of research. Classifying the environment surrounding certain sounds such as offices, parks, and busses are ASR, and classifying and recognizing certain temporal sound events are AER. ESC is classifying the sounds and environment hence comes under ASR.

Classifying the sounds in the environment-like glass breaking, gunshots, and dog barking can help applications such as audio surveillance [9] and home automation

A. Bansal (✉) · N. K. Garg
GZSCCET, MRSPTU, Bathinda, India
e-mail: anambansal19@gmail.com

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_52

[46]. Smart homes can be designed to assist elderly people staying at home [36]. ESC can help in the development of robotic navigation systems [3]. Monitoring wildlife such as bats [23], frogs [8], and birds [44] sound classification is applications of ESC.

Environmental sounds are complex to be processed as these are different as compared to speech and music. Environmental sounds do not have a particular structure and have a low signal-to-noise ratio [26]. Researchers have also used video cameras for environmental sound recognition(ESR) but microphones have advantages as compared to video cameras as the former are omnidirectional and are not affected by occlusion [4].

Different aspects of ESC are explored in past studies. The most recent studies focus on ESC using spectrogram along with data augmentation [26] and transfer learning [21]. The phases involved in ESC are data collection, preprocessing, feature extraction and feature selection, and classification of the data (Fig. 1). The data is either self-collected or standard dataset is used. The data need to be preprocessed to remove the noise artifacts. Further, the sound-relevant features are extracted and important features which help in classification are selected. Lastly, the algorithms for classification are applied to classify the sounds.

The main contributions of this paper are:

1. Determining the efficacy of ensemble model Adaboost for ESC.
2. Testing various hyperparameters of Adaboost for ESC.
3. Comparing the accuracy of various base estimators used in Adaboost.

The remaining paper is structured as follows: In Sect. 2, a survey related to ESC is demonstrated. In Sect. 3, the methodology and system overview of the applied approach are described. In Sect. 4, various experiments and their results are listed. In Sect. 5, the conclusion and future work are described.

2 Literature Review

Many researchers have worked toward the environmental sound classification.

Most of the researchers have used publicly available datasets for environmental sound classification. The datasets that have been used in various studies and are publicly available in ESC-10 [29], ESC-50 [29], and UrbanSound8k [34]. ESC-10 dataset consists of 400 recordings of 10 classes (dog bark, rain, sea waves, baby cry, clock tick, person sneeze, helicopter, chainsaw, rooster, and fire crackling). Each class has 40 recordings and each recording is of 5 s. ESC-50 dataset comprises 2000 recordings from 50 different classes. 50 classes are placed in 5 major groups— animals, natural soundscapes and water sounds, human non-speech sounds, interior/domestic sounds, and exterior/urban noises. Each class has 40 recordings of 5 s each. ESC-10 is a subset of ESC-50. UrbanSound8k dataset consists of 8732 labeled sound recordings from 10 classes—air conditioner, car horn, children playing, dog bark,

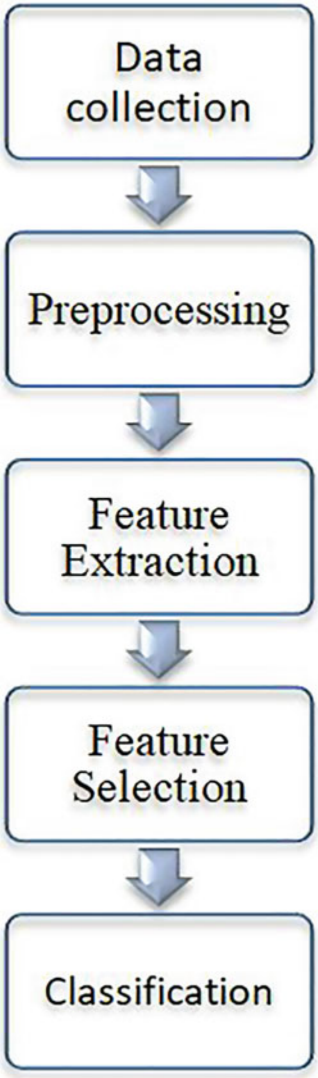


Fig. 1 Stages of environmental sound classification

drilling, engine idling, gunshot, jackhammer, siren, and street music. The duration of each recording is less than 4 s. These sounds from the free sound repository [13] are filtered and labeled manually to generate the UrbanSound8k dataset for ESR. There are diverse self-collected datasets that have been used for one-time research.

The dataset that is acquired is not amenable to being used immediately. It needs to be preprocessed to remove the noise or reduce the dimensions. Amplitude-based silence detection technique [27], dimensionality reduction [31, 49], and other preprocessing techniques [51, 54] are used in past studies for environmental sound classification to enhance the signals.

Audio features can be broadly categorized into cepstral features, spectral features, temporal features, and image-based features. Various researchers used different kinds of features in their works or a combination of different categories of features. Temporal features are the time domain features directly extracted from audio signals such as zero crossing rate [11], autocorrelation [47], and linear predictive coefficients [41]. Spectral features are obtained by exposing temporal features to some transformations such as discrete chiplet transform, discrete curvelet transform, and discrete Hilbert transform along with fast Fourier transforms [16]. Various other spectral features such as spectral flatness [11], spectral dynamic Features [18], spectral contrast [15], and MPEG-7 audio feature sets [50] [27] are used for ESC. Cepstral features consist of mel-frequency cepstral coefficients(MFCCs) [11, 15, 27, 37, 38, 54] which are extensively used in ESC. Variation of MFCCs called independent component analysis transformed MFCCs is also used in one of the past studies [51]. Certain image-based features are used by researchers, and considerable accuracy is obtained as compared to other features. Spectrogram and cross recurrence plots are used [6, 54]. Log mel spectrogram(LMS) performs better as compared to MFCCs [2, 19]. In a study [10], LMS is combined with log gammatone features to achieve good accuracy of 83.3%. There are certain other domain features used in ESC like haar-like sound features inspired by haar-like filtering in the case of 2D face detection [53],constant-Q transform (CQT) features [24], pitch range(PR)-based feature set [45], and gammatone wavelet features [48]. Handcrafted features are combined with deep features for ESC, and an accuracy of 96.16% is obtained [22].

Not all the extracted features are important for ESC. So, a set of features need to be selected so that they can be computationally less complex [20]. The feature sets that do not contribute to classification are correlated and can be discarded. Different feature sets are prepared and used for ESC [7]. The principal component analysis is one of the techniques that can be used for feature selection in ESC [30]. Feature vector sets are used instead of feature sets for ESC [30].

A variety of classifiers are used by researchers for ESC. The main categories of classifiers for ESC are traditional machine learning classifiers and deep learning classifiers. Various machine learning classifiers for ESC are compared in the past [17, 39]. Support vector machine (SVM) [42, 56], K-nearest neighbor (KNN) [11], Gaussian mixture model(GMM) [5, 25], hidden Markov model (HMM) [7], and artificial neural network(ANN) [7] are few machine learning classifiers used for ESC. Both one-class SVM [30] and multiclass SVM [51] are used for ESC. Kernels of SVM can be varied to attain good accuracy is ESC [45]. KNN varies the value of k to get considerable accuracy [7]. HMM attains higher accuracy and consumes less power as compared to other machine learning classifiers [53].

Deep neural networks (DNNs) are also used in ESC. Various DNNs used are convolutional neural networks (CNNs) [14, 55], convolutional recurrent neural networks (CRNNs) [35], deep belief neural networks (DBNNs) [15], tensor deep stacking neural networks (TDSNNs) [19], and image recognition networks [6, 52]. The first-ever use of CNN in ESC was done by Piczak [28]. The hyperparameters of CNN are varied to get considerable accuracy for ESC. Experiments are performed with various activation functions (ReLu, PReLu, SoftPlus, LeakyReLu, and ELU) [56]. The number of layers of CNN is varied to find the effect of changing the number of layers. In a study [12], very deep CNN with 34 weight layers gave around 15% more accuracy than two-layered CNN. CNN can be sequential, parallel, and end-to-end [1], and parallel CNN gives the best accuracy [24]. CNNs give great accuracy improvement when it works with augmented data [33].

Certain other classifiers such as self-organizing maps [40]and Bayesian belief networks [43] are studied by researchers for ESR.

3 Methodology

3.1 Dataset

The experiments are conducted on UrbanSound8K dataset which is available publicly. It contains 8732 sound recordings labeled into ten classes. The class labels and corresponding number of recordings for each class are shown in Table 1.

3.2 System Overview

Feature Extraction: Features are characteristics of the sounds that classify environmental sounds. 40 mel-frequency cepstral coefficients (MFCCs) are extracted for each audio file. The extracted features are split into train and test sets.

Table 1 Details of dataset

Class name	Number of recordings	Class label
Air conditioner	1000	0
Car horn	429	1
Children playing	1000	2
Dog bark	1000	3
Drilling	1000	4
Engine idling	1000	5
Gunshot	374	6
Jackhammer	1000	7
Siren	929	8
Street music	1000	9

Classification: Adaboost algorithm is used for classification. It is a boosting technique that uses decision trees for modeling. In Adaboost, sequence of models is created, and each model corrects the errors from the last model. Weights are assigned to the incorrectly predicted observations, and these values are predicted correctly by the succeeding model.

Parameters used in the algorithm:

- **base_estimators**: The machine earning algorithm used to fit into a random subset of the dataset. By default, decision tree is used as the base estimator
- **n_estimators**: The number of base estimators is specified by this parameter. There is always trade off. The value for this parameter needs to be set accordingly. The higher value will increase the running time, and the lower value may not lead to good accuracy. The default value is 50.
- **learning_rate**: The contribution of the base estimators in the final combination is controlled by the learning rate. The default value is 1.
- **max_depth**: This parameter describes the maximum depth of each base estimator.
- **random_state**: This parameter is an integer value. It defines the random data split.
- **n_jobs**: This parameter specifies the number of processors that can be used by algorithm. This value is set to -1 so that maximum processors are used.

In this paper, experiments are performed by tuning the parameters such as base_estimators, n_estimators, and learning_rate. The results are compared with the results of machine learning algorithms (Support Vector Machines(SVM), IBK5, RandomForest500, J48, and ZeroR) used in benchmark research [34].

4 Experimental Setup and Results

The dataset is split into a training size of 0.7 and a test set size of 0.3. The random state is fixed to 96. By default, a decision tree is used as base estimators. The n_estimator has a default value of 10, and the learning rate has a default value of 1.

4.1 Tuning the Parameters

Experiments performed for tuning various parameters are described below:

- **Base Estimator**: The experiments are performed with three machine learning algorithms—decision tree(default), random forest, support vector classifier. Random forest classifier provides an accuracy of 88.44% with other parameters having default values. Support vector classifier(SVC) offers an accuracy of 30.8%, and the default decision tree model provides an accuracy of 45.76%. Figure 2 depicts the

Fig. 2 Effect of base estimators

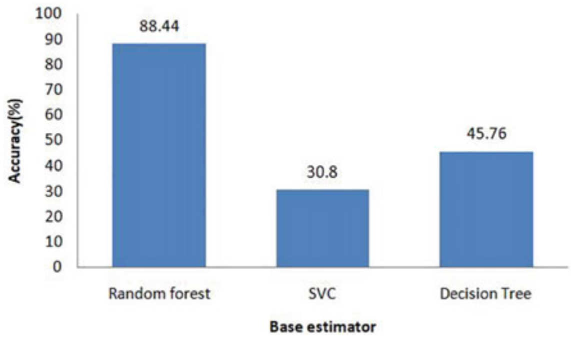
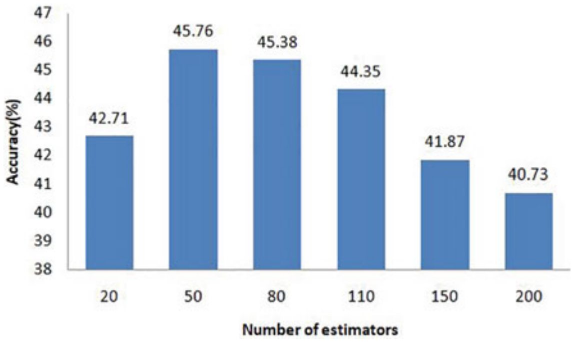


Fig. 3 Effect of base estimators



effect of different base estimators. Since decision tree classifier is the default classifier, so all the remaining experiments are performed keeping the base estimator as the decision tree.

- **Number of Estimators**: Experiments are performed with number of estimators as 20(42.71%), 50(45.76%), 80(45.38%), 110(44.35%), 150(41.87%), and 200(40.73%). So an adequate number of estimators need to be selected to achieve desirable accuracy. The effect of the number of estimators is shown in Fig. 3.
- **Learning Rate**: Learning rate is varied—0.01(30.72%), 0.1(43.47%),0.5(49.42%), 0.75(48.36%),0.99(45.01%), and 1(45.76%). So, adequate learning rate in this case is 0.5. Figure 4 demonstrates the effect of learning rate on accuracy.

4.2 Comparison with Results of Benchmark Research

Table 2 shows the comparison of the Adaboost algorithm with the machine learning algorithms used in benchmark research. Also refer to Fig. 5.

Fig. 4 Effect of learning rate

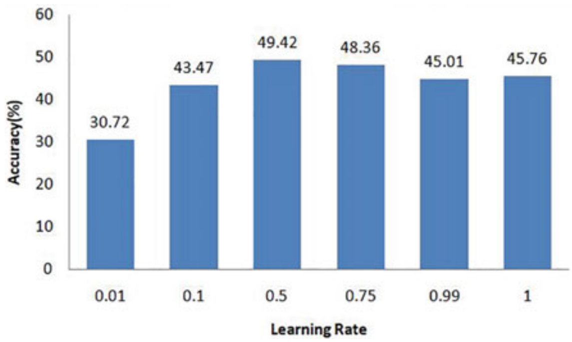
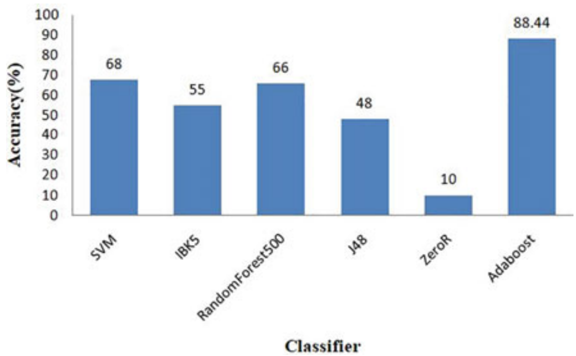


Table 2 Comparison of accuracy with benchmark research

Algorithm	Accuracy (%)
SVM [34]	68
IBK5 [34]	55
RandomForest500 [34]	66
J48 [34]	48
ZeroR [34]	10
Adaboost	88.44

Fig. 5 Comparison with benchmark classification results



5 Conclusion

In this paper, ensemble boosting technique—Adaboost— is used for classifying the environmental sounds. UrbanSound8k dataset is used, and sounds are classified into ten categories. Few parameters are tuned for the dataset. Experiments are performed for various base estimators, learning rate, and a number of estimators. It has been found that random forest gives the highest accuracy (88.44%) as compared to a decision tree (45.76%) and support vector classifier (30.8%). The learning rate of 0.5 and 50 estimators provides accurate results. Further, the other parameters can be tuned [32] and performance can be improved. Although, a plethora of researches has been done using deep learning models for ESC, transfer learning can be employed in the future with better accuracy.

References

1. Abdoli S, Cardinal P, Koerich AL (2019) End-to-end environmental sound classification using a 1D convolutional neural network. Expert systems with applications 136:252–263
2. Ahmed M, Robin TI, Shafin AA et al (2020) Automatic environmental sound recognition (AESR) using convolutional neural network. Int J Mod Educ Comput Sci 12(5)
3. Aziz S et al (2019) Automatic scene recognition through acoustic classification for behavioral robotics. Electronics 8(5):483
4. Bansal A et al (2018) An off the shelf CNN features based approach for vehicle classification using acoustics. In: International conference on ISMAC in computational vision and bio-engineering. Springer, pp 1163–1170
5. Barchiesi D et al (2015) Acoustic scene classification: classifying environments from the sounds they produce. In: IEEE Signal Process Mag 32(3):16–34
6. Boddapati V et al (2017) Classifying environmental sounds using image recognition networks. Proc Comput Sci 112:2048–2056
7. Bountourakis V, Vrysis L, Papanikolaou G (2015) Machine learning algorithms for environmental sound recognition: towards soundscape semantics. In: Proceedings of the audio mostly 2015 on interaction with sound, pp 1–7
8. Brodie S et al (2020) Automated species identification of frog choruses in environmental recordings using acoustic indices. Ecol Indicators 119:106852
9. Chandrakala S, Jayalakshmi SL (2019) Environmental audio scene and sound event recognition for autonomous surveillance: a survey and comparative studies. ACM Comput Surveys (CSUR) 52(3):1–34
10. Chi Z, Li Y, Chen C (2019) Deep convolutional neural network combined with concatenated spectrogram for environmental sound classification. In: 2019 IEEE 7th international conference on computer science and network technology (ICCSNT). IEEE, pp 251–254
11. Chu S et al (2006) Where am I? Scene recognition for mobile robots using audio features. In: 2006 IEEE international conference on multimedia and expo. IEEE, pp 885–888
12. Dai W et al (2017) Very deep convolutional neural networks for raw waveforms. In: 2017 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE. pp 421–425
13. Font F, Roma G, Serra X (2013) Freesound technical demo. In: Proceedings of the 21st ACM international conference on multimedia, pp 411–412
14. Garg S et al (2021) Urban sound classification using convolutional neural network model. IOP Conf Ser Mater Sci Eng 1099:012001
15. Gencoglu O, Virtanen T, Huttunen H (2014) Recognition of acoustic events using deep neural networks’. In: 2014 22nd European signal processing conference (EUSIPCO). IEEE, pp 506–510
16. Han B, Hwang E (2009) Environmental sound classification based on feature collaboration’. In: 2009 IEEE international conference on multimedia and expo. IEEE, pp 542–545
17. Jekic N, Pester A (2018) Environmental sound recognition with classical machine learning algorithms. In: International conference on remote engineering and virtual instrumentation. Springer, pp 14–21
18. Karbasi M, Ahadi SM, Bahmanian M (2011) Environmental sound classification using spectral dynamic features. In: 2011 8th international conference on information, communications & signal processing. IEEE, pp 1–5

19. Khamparia A et al (2019) Sound classification using convolutional neural network and tensor deep stacking network. IEEE Access 7:7717–7727
20. Liu H et al (2010) Feature selection: an ever evolving frontier in data mining. In: Feature selection in data mining. PMLR, pp 4–13
21. Lu J et al (2021) Deep convolutional neural network with transfer learning for environmental sound classification. In: 2021 international conference on computer, control and robotics (ICCCR). IEEE, pp 242–245
22. Luz JS et al (2021) Ensemble of handcrafted and deep features for urban sound classification. Appl Acoust 175:107819
23. Mac Aodha O et al (2018) Bat detective? Deep learning tools for bat acoustic signal detection. PLoS Comput Biol 14(3):e1005995
24. Mendoza JM et al (2018) Audio event detection using wireless sensor networks based on deep learning. In: International wireless internet conference. Springer, pp 105–115
25. Muhammad G et al (2010) Environment recognition using selected MPEG-7 audio features and mel-frequency cepstral coefficients. In: 2010 fifth international conference on digital telecommunications. IEEE, pp 11–16
26. Mushtaq Z, Su S-F, Tran Q-V (2021) Spectral images based environmental sound classification using CNN with meaningful data augmentation. Appl Acoust 172:107581
27. Ntalampiras S, Potamitis I, Fakotakis N (2010) Automatic recognition of urban environmental sounds events
28. Piczak KJ (2015) Environmental sound classification with convolutional neural networks. In: 2015 IEEE 25th international workshop on machine learning for signal processing (MLSP). IEEE, pp 1–6
29. Piczak KJ (2015) ESC: dataset for environmental sound classification. In: Proceedings of the 23rd ACM international conference on multimedia, pp 1015–1018
30. Rabaoui A et al (2008) Using one-class SVMs and wavelets for audio surveillance. IEEE Trans Inf Forensics Secur 3(4):763–775
31. Ragab MG et al (2021) An ensemble one dimensional convolutional neural network with Bayesian optimization for environmental sound classification. Appl Sci 11(10):4660
32. Raval P, Christopher J (2021) Parameter tuning for wavelet-based sound event detection using neural networks. In: International conference on computational intelligence in music, sound, art and design (part of EvoStar). Springer, pp 235–247
33. Salamon J, Bello JP (2017) Deep convolutional neural networks and data augmentation for environmental sound classification. IEEE Signal Process Lett 24(3):279–283
34. Salamon J, Jacoby C, Bello JP (2014) A dataset and taxonomy for urban sound research. In: Proceedings of the 22nd ACM international conference on multimedia, pp 1041–1044
35. Sang J, Park S, Lee J (2018) Convolutional recurrent neural networks for urban sound classification using raw waveforms. In: 2018 26th European signal processing conference (EUSIPCO). IEEE, pp 2444–2448
36. Saraubon K, Anurugsa K, Kongsakpaibul A (2018) A smart system for elderly care using IoT and mobile technologies. In: Proceedings of the 2018 2nd international conference on software and EBusiness. ICSEB'18. Association for Computing Machinery, Zhuhai, China, pp 59–63. ISBN: 9781450361279. DOIurl10.1145/3301761.3301769
37. Sharan RV, Moir TJ (2019) Acoustic event recognition using cochleagram image and convolutional neural networks. Appl Acoust 148:62–66
38. Sigtia S et al (2016) Automatic environmental sound recognition: performance versus computational cost. IEEE/ACM Trans Audio Speech Lang Process 24(11):2096–2107
39. Silva BD et al (2019) Evaluation of classical machine learning techniques towards urban sound recognition on embedded systems. Appl Sci 9(18):3885
40. Sitte R, Willets L (2007) Non-speech environmental sound identification for surveillance using self-organizing-maps. In: Proceedings of the fourth conference on IASTED international conference: signal processing, pattern recognition, and applications, pp 281–286
41. Tak RN, Agrawal DM, Patil HA (2017) Novel phase encoded mel filterbank energies for environmental sound classification. In: International conference on pattern recognition and machine intelligence. Springer, pp 317–325

42. Theodorou T, Mporas I, Fakotakis N (2015) Automatic sound recognition of urban environment events. In: International conference on speech and computer. Springer, pp 129–136
43. Tsau E, Kim S-H, Jay Kuo C-C (2011) Environmental sound recognition with CELP-based features. In: ISSCS 2011-international symposium on signals, circuits and systems. IEEE, pp 1–4
44. Tuncer T, Akbal E, Dogan S (2021) Multileveled ternary pattern and iterative ReliefF based bird sound classification. Appl Acoust 176:107866
45. Uzkent B, Barkana BD, Cevikalp H (2012) Non-speech environmental sound classification using SVMs with a new set of features. Int J Innov Comput Inf Control 8(5):3511–3524
46. Vafeiadis A et al (2017) Audio-based event recognition system for smart homes. In: 2017 IEEE SmartWorld, ubiquitous intelligence & computing, advanced & trusted computed, scalable computing & communications, cloud & big data computing, internet of people and smart city innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI). IEEE, pp 1–8
47. Valero X, Alías F (2012a) Classification of audio scenes using narrow-band autocorrelation features. In: 2012 Proceedings of the 20th European signal processing conference (EUSIPCO). IEEE
48. Valero X, Alías F (2012b) Gammatone wavelet features for sound classification in surveillance applications. In: 2012 Proceedings of the 20th European signal processing conference (EUSIPCO). IEEE, pp 1658–1662
49. Van Der Maaten L, Postma E, Van den Herik J (2009) Dimensionality reduction: a comparative. J Mach Learn Res 10(66–71):13
50. Wang J-C et al (2006) Environmental sound classification using hybrid SVM/KNN classifier and MPEG-7 audio low-level descriptor. In: The 2006 IEEE international joint conference on neural network proceedings. IEEE, pp 1731–1735
51. Wang J-C et al (2008) Robust environmental sound recognition for home automation. IEEE Trans Autom Sci Eng 5(1):25–31
52. Yao K et al (2014) Robust deep feature extraction method for acoustic scene classification. In: 2019 IEEE 19th international conference on communication technology (ICCT). IEEE, pp 198–202
53. Zhan Y, Kuroda T (2014) Wearable sensor-based human activity recognition from environmental background sounds. J Amb Intell Human Comput 5(1):77–89
54. Zhang H, McLoughlin I, Song Y (2015) Robust sound event recognition using convolutional neural networks. In: 2015 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, pp 559–563
55. Zhang K et al (2020) Environment sound classification system based on hybrid feature and convolutional neural network. In: Xibei Gongye Daxue Xue bao/J Northwestern Polytech Univ 38(1):162–169
56. Zhang X, Zou Y, Shi W (2017) Dilated convolution neural network with LeakyReLU for environmental sound classification. In: 2017 22nd international conference on digital signal processing (DSP). IEEE, pp 1–5

Blockchain-Based Intelligent Agreement for Healthcare System: A Review

Anu Raj, Shiva Prakash, Jyoti Srivastva, and Rajkumar Gaur

Abstract A Blockchain-based smart contract is widely used in every domain for secure data exchange and data storage. The latest technology operates automatically, controls, or documents legally relevant events and actions according to the agreements described in the contract agreement. The Blockchain in health care can be visualized in managing electronic medical record (EMR) data, preservation of healthcare data, personal health record data control, point-of-care genomics supervision, computerized health reports data control, etc., by using IoT devices for data collections. This paper presents a brief overview of the well-known existing researches based on Blockchain-enabled intelligent contracts in the healthcare system. The paper focuses on existing research, methodologies, and future trends and comparative analysis of smart contracts (intelligent agreement) methods . This paper points out challenges and open problems that require discussion in the future considerations. Moreover, help new researchers to understand the upcoming trends in Blockchain-based intelligent agreements in the healthcare scheme.

Keywords Blockchain · Healthcare system · Personal health records · Internet of things · Intelligent agreements

1 Introduction

Nowadays, Blockchain is widely used in every domain for secure data exchange as well as data storage. It is a technology that has an immutable ledger for storing data and tracking assets, etc. It strengthens the e-health applications securities, monitoring

A. Raj (✉) · S. Prakash · J. Srivastva · R. Gaur
ITCA, MMMUT, Gorakhpur, Uttar Pradesh, India
e-mail: anu.raj10@yahoo.com

S. Prakash
e-mail: spitca@mmmut.ac.in

R. Gaur
e-mail: rajkumargaur@mmmut.ac.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_53

devices security, recording and exchanging e-media records, medical trial data, and remote health monitoring records [1, 12]. It provides a secure platform to every domain to secure data exchange and its management using smart contracts. It has an extensive utilization area, such as business services, pharmaceutical services, forecast markets, and the Internet of Things (IoT). Particular utilization of this technology in the field of the medical care system is essentially based on the intelligent agreement, which depends on the clinical work supervision and is restrained by the framework. Enhancement of utilization is the way to framework updating [3]. With the use of Blockchain, the issues of the lack of interoperability among various medical services providers can be tackled, bringing down task expenses and coordination efforts. Also, patients' health records integrity is guaranteed and permits them to possess their private health records [6, 7]. The collaboration network connected by Blockchain may perform an essential role in value creation, exchanging, and sharing data between the medical services community. This technology is one of the newest innovations to clarify security, assurance, and clarity in the medical sector.

Protection, separation, and scalability of health records storage and sharing are necessary for investigation in healthcare and clinical centers. The health center's health inquiries about the patient's fitness depend on their scientific data [17]. First of all, data collected by wearable devices is first sent to the cloud over the Internet for processes [18]. The Blockchain constancies for healthcare data direction design to utilities for sufferers, specialists, and healthcare history access institutes inpatient and control, requests and cash management, IoMT security management, and analysis of data affirmation and replacement for clearness of financial audit [25]. For specific purposes, consecutive updates and encoded, decentralized blockchain histories are made to recognize, monitor, and handle clinical records [23]. Blockchain investigates the healthcare domain [16]. This deliberate review combines research that introduces a new healthcare solution and methodology, which resolves preservation and secrecy concerns. It builds trust between patients and healthcare providers.

This article constructs as follows: Sect. 2 represents a brief description of related work. The blockchain-based intelligent contracts for the healthcare system and compares the existing research method involved in blockchain. Section 3 presents the detail about the operational process of smart contracts, platform, and future trends of the smart contract. Section 4 describes the open issues and research challenges. Lastly, Sect. 5 concludes the paper.

2 Literature Review

This section presents a brief overview of well-known pieces of knowledge that have considered blockchain-enabled intelligent agreements in the healthcare system. Some of them are as follows:

Gordon and Catalini describe healthcare blockchain technology [8]. They ended their analysis on how this technology can allow patient-centric healthcare data sharing over the institution-centric direction. Asma Khatoon introduced multiple workflows in the healthcare system using blockchain technology for better data management [11]. Tim K. Mackey et al. explained blockchain approaches that reacted to the latest difficulties encountered in the medical sector associated with other economic sectors [13]. So, this reason, assuring that a health blockchain is 'fit-for-purpose' central. These medical care centers share the multidisciplinary procedures construction and deployment. This proposed application of the Blockchain presents reliable management and research of crucial healthcare data. However, blockchains are computationally significant, require particular bandwidth, and more computational power-saving [4]. So, the proposed four blockchain-based intelligent contracts for user affirmation, entrance authorization, fault disclosure, and access cancelation. In this design, identifying with the block area of the ledger and a massive quantity of patient records, the EMRs are managed in the cloud after transfer encrypted through cryptographic techniques like ECC and EdDSA. At the same time, their identical hashes are organized into the Blockchain. Wang et al. [19] presented a comprehensive overview of Blockchain-powered intelligent contracts. Pradhan et al.[22] introduced a solidity-based innovative agreement that treated up an Ethereum pragmatic machine to promote responsible analysis and store events from the sensors. Fatma Ellouze1 et al. performed a scientific analysis of present solutions by implementing blockchain technology on IoMT [15]. In the distributed problems concern the application of Blockchain over IoMT and also specified techniques. Shafaq Naheed Khan et al. performed a complete survey of blockchain-enabled innovative recognition from mechanical and acceptance points [5]. Griggs et al. [10] introduced the applicability of blockchain-based intelligent agreements to promote reliable review and limitation of pharmaceutical sensors. This innovative compressed method would improve real-time patient monitoring and medical interferences. It transfers information to victims and medical experts while managing a reliable record of receiving these actions. Manaf Zghaibeh et al. performed Smart-Health, a blockchain-based health authority scheme [9]. It separates multi-layered Blockchain mixed with a multi-layer address method that determines the prerogatives and recognition of things in the scheme. Ashutosh Sharma et al. presented the concept of Blockchain and intelligent agreements and their applications on the Internet of Medical Things (IoMT) and the e-healthcare sectors [20, 24].

2.1 Comparison and Analysis

In this section, there is a brief comparison of various research methods involved in blockchain, their objective, description, and their corresponding authors, which are as follows:

Researchers have advised in literature reviews to report on the current problems. The main work of similar analysis techniques of blockchain is listed in Table 1. It shows the comparison among well-known existing technologies in the healthcare system. The above table finds that most of the authors tried to resolve the security issues of the healthcare system and patients' data management. In this study, we

Table 1 Comparison of existing blockchain-based researches

Authors and years	Research method	Objective	Description
Maslove et al. [14]	Proof-of-concept	Clinical trials data management	The proposed method improved the integrity of data obtained during clinical examination, serving researchers, controls, and drug corporations identical
Chen et al. [2]	System design	Secure medical data storage and service framework	A storage design was intended to maintain personal medical data based on blockchain and cloud warehouse without third parties
Zhang et al. [26]	Framework construction	Development of balanced scorecard evaluation framework	A structure was intended to evaluate the appearance of blockchain ambitions in implementing value-based holistically
Shuaib et al. [21]	Literature review	Blockchain potential in improving secured digitized medicine	Account tracks record what activities and alterations are performed to patient reports while informing all users on the system
Khatoon [11]	Framework construction	Smart contract	This work introduced various workflows connected with the healthcare ecosystem using blockchain techniques for more reliable data supervision
Du et al. [3]	System design	fuzzy set theory to filter unnecessary attributes	These decision-making methods are experimental. This method manages the various aspects as characteristics and interpretive structure modeling
Saini et al. [19]	System design	A smart contract-based access control framework for cloud smart healthcare system	To build an access control structure based on innovative agreements built on the top of a shared ledger (blockchain) system to defend the distribution of EMRs, among various things included in the original healthcare scheme

addressed the practical considerations while building a healthcare blockchain and innovative contract system. Specifically, we compared the technical features of the blockchain as well as the objective of the research.

3 Intelligent Agreements

An intelligent agreement is defined as a mutual agreement between pair or added persons or organizations that contain data, processes input variables. It gives outputs as the pre-determined conditions meet. It supports the pre-determined function that empowers intelligent contract generation. The operation involved in a blockchain-based smart contract, platform, and future trends of the intelligent contract are as follows:

On the basis of the above table, Fig. 1 has been drawn which include the type of schemes such as system design, proof of concept, literature review, and framework construction. It shows that most of the work has been done on designing the system using smart contract to facilitate, execute, and enforce contract between unauthorized person without the consent of authorized party.

3.1 Operational Techniques of Intelligent Agreements

An intelligent agreement is probably like a company that includes essential variables, structures, events, and functional modifiers, which are required to achieve and control appropriate actions and events as per the deal cycles. It can also call other intelligent agreements.

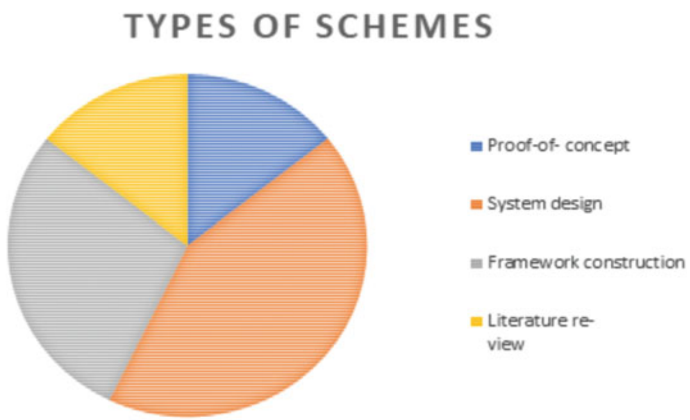


Fig. 1 Graphical representation of comparative study of blockchain-based research

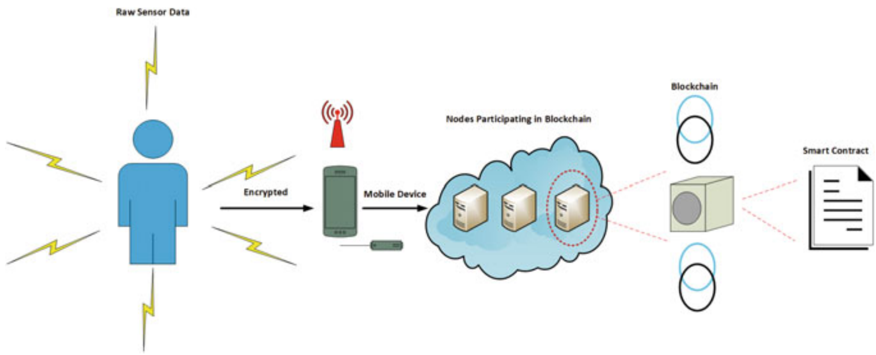


Fig. 2 Blockchain-based smart contract managing different permissions in healthcare system

Every intelligent contract is incorporated with times and functions. It is usually stored within a section that could be assigned for all conditions from manage various authorizations to access the information, as shown in Fig. 2. It can be viewed that various partners are joined with this plan and perform various activities. This will assist make more conventional co-operations with the victims and the healthcare providers.

The data obtaining permission rule is also joined in the intelligent agreement that benefits in tracing every performance of the patients by applying the unique IDs produced during the patients' registrations.

3.2 Intelligent Agreements Platforms

Intelligent agreements can be created and expanded in various blockchain policies such as Hyperledger Fabric, Bitcoin, solidity, multichain, etc. High-level programming languages are supported by most of the platforms to develop the smart contract. However, Bitcoin is the primary blockchain platform to handle digital business exchanges with a highly restricted computing capacity. NXT is an open-source blockchain policy that depends totally on a proof-of-stake agreement convention. Solidity is also a high-level programming language that utilizes to create smart contract and agreement codes. It is compiled in Ethereum virtual machine and expanded on the blockchain for execution. Ethereum is the most popular platform for innovative agreements. It can be appropriated to produce various types of decentralized purposes (DApps) in different regions. Hyperledger Fabric trust just a variety of business-related associations that participate by enrollment service providers. Multichain is the most accessible blockchain platform in terms of setup. It is designed to be a permissioned blockchain that is very simple yet powerful to use and inherits proven features of the famous Bitcoin Blockchain by forking from it.

3.3 Future Trends

This subsection performs the future trends of the blockchain technologies-based intelligent contract in the healthcare system. It explains the forthcoming perspective of the intelligent contract in healthcare operations which are as in Table 2 follows.

Table 2 Comparison of existing Blockchain-based researches

S. No.	Smart contract methods	Description
1	Layer 2 protocol	Layer two is an overlay network that rests on the peak of the underlay blockchain. The primary objective of the layer two protocols is to tackle the data sharing activity and scaling provocations faced by significant cryptography systems
2	Contract management solutions	It is as of now imaginable to utilize intelligent contracts even without blockchain. Hence, deal control solutions can resolve both changeless effects and the immutable nature of blockchain technology. It takes care of the agreement's life cycle while eliminating limitations on the existing technology
3	Fabasoft contracts	Fabasoft contracts is one of the most modern intelligent agreement solutions that are made to utilize cloud-based programming. It helps clients throughout the whole agreement life cycle from cross-organization contract planning, skillful audit, and permission measures, and the alteration of stable agreement archives
4	Parallel blockchain	The agreement nodes and intelligent agreements in the blockchain will create decentralized autonomous corporations and decentralized autonomous organizations participating in a distinctive kind of dapp. Form DAS corresponds to the counterfeit society ACP. The programmable element of innovative agreement authorizes the blockchain to implement various kinds of innovative plans and situation origin. It results from the assessment, through which the optimal determination can be obtained either semi-automatically or fully automatically

4 Research Challenges and Open Issues

Blockchain-based intelligent agreements are an emerging technology but face many challenges. These challenges are different types as legal issues, reliance on "off-chain" resources, immutability, dependability, scalability, and agreement mechanism. Some of the listed issues are as follows:

4.1 Legal Issues

The legitimate concern of intelligent agreements is one of the crucial aspects of intelligent commitment difficulties, which include three-part: (i) every nation has its laws and guidelines; subsequently, it is convoluted to guarantee consistency will all guidelines, (ii) law statements or opinions are not quantifiabl, and (iii) legislatures are strong on a directed and managed utilization of the blockchain change in various purposes.

4.2 Reliance on "Off-Chain"

Resources-based intelligent agreements require getting data or information from devices, not the blockchain individual, assumed off-chain devices. For this purpose, oracles are confided in outsider somebody that recover off-chain data and drive that data to the blockchain by pre-determined periods.

4.3 Immutability Issue

One more constraint in the blockchain individual that moves intelligent agreements is the immutable aspect of the blockchain. Once this agreement is created, it cannot be modified.

4.4 Consensus Mechanism Issue

The agreement system supports determining the security issues and provides the flexibility of decentralization in the blockchain networks system. There are a few existing agreement protocols, including, proof-of-stake, Proof-of-Authentication Proof-of-Work. Though the PoW protocol empowers security in the blockchain, it squanders assets.

5 Conclusion

In this article, perform a complete review of blockchain-enabled intelligent agreements for the healthcare scheme. This paper contributes a brief overview of the well-known existing researches based on blockchain-enabled intelligent agreements in healthcare policy. Comparison is also made of these existing innovative contract-based researches in the healthcare system. Consequently, blockchain technologies in healthcare directions involve things and somebody to enhance their experience with secrecy and authenticity. Also, it explores the latest challenges and open issues that the intelligent deal and performed its expected trends. The design to conduct additional analysis on identical blockchain and associated intelligent contract applications. This research presents informational provisions to involved researchers to work on blockchain-based smart contracts.

References

1. Alharby M, Van Moorsel A (2017) Blockchain-based smart contracts: a systematic mapping study. arXiv:1710.06372
2. Chen Y, Ding S, Xu Z, Zheng H, Yang S (2019) Blockchain-based medical records secure storage and medical service framework. J Med Syst 43(1):1–9
3. Du X, Chen B, Ma M, Zhang Y (2021) Research on the application of blockchain in smart healthcare: constructing a hierarchical framework. J Healthcare Eng 2021
4. Dwivedi AD, Srivastava G, Dhar S, Singh R (2019) A decentralized privacy-preserving healthcare blockchain for iot. Sensors 19(2):326
5. Ellouze F, Fersi G, Jmaiel M (2020) Blockchain for internet of medical things: a technical review. In: International conference on smart homes and health telematics. Springer, pp 259–267
6. Fonseca Ribeiro MID, Vasconcelos A (2020) Medblock: using blockchain in health healthcare application based on blockchain and smart contracts. ICEIS 1:156–164
7. Gaur R, Prakash S (2021) Performance and parametric analysis of iot's motes with different network topologies. In: Innovations in electrical and electronic engineering. Springer, pp 787–805
8. Gordon WJ, Catalini C (2018) Blockchain technology for healthcare: facilitating the transition to patient-driven interoperability. Comput Struct Biotechnol J 16:224–230
9. Griggs KN, Ossipova O, Kohlios CP, Baccarini AN, Howson EA, Hayajneh T (2018) Healthcare blockchain system using smart contracts for secure automated remote patient monitoring. J Med Syst 42(7):1–7
10. Khan SN, Loukil F, Ghedira-Guegan C, Benkhelifa E, Bani-Hani A (2021) Blockchain smart contracts: applications, challenges, and future trends. Peer-to-peer networking and applications, pp 1–25
11. Khatoon A (2020) A blockchain-based smart contract system for healthcare management. Electronics 9(1):94
12. Kumar R, Gupta D (2011) Security in real time multimedia data based on generalized keys. In: Proceedings of the international conference on advances in computing and artificial intelligence, pp 93–96
13. Mackey TK, Kuo TT, Gummadi B, Clauson KA, Church G, Grishin D, Obbad K, Barkovich R, Palombini M (2019) 'Fit-for-purpose?'-challenges and opportunities for applications of blockchain technology in the future of healthcare. BMC Med 17(1):1–17

14. Maslove DM, Klein J, Brohman K, Martin P (2018) Using blockchain technology to manage clinical trials data: a proof-of-concept study. JMIR Med Inform 6(4):e11949
15. Pradhan NR, Rout SS, Singh AP (2021) Blockchain based smart healthcare system for chronic—illness patient monitoring. In: 2020 3rd international conference on energy, power and environment: towards clean energy technologies. IEEE, pp 1–6
16. Raj A, Prakash S (2018) Internet of everything: a survey based on architecture, issues and challenges. In: 2018 5th IEEE Uttar Pradesh section international conference on electrical, electronics and computer engineering (UPCON). IEEE, pp 1–6
17. Raj A, Prakash S (2019) Mobile data gathering approaches in wireless sensor networks: a survey. In: 2019 6th international conference on computing for sustainable global development (INDIACom). IEEE, pp 758–762
18. Raj A, Prakash S (2020) Path discovery approach for mobile data gathering in wsn. Int J Comput Appl Technol 64(2):133–142
19. Saini A, Zhu Q, Singh N, Xiang Y, Gao L, Zhang Y (2020) A smart-contract-based access control framework for cloud smart healthcare system. IEEE Internet Things J 8(7):5914–5925
20. Sharma A, Tomar R, Chilamkurti N, Kim BG et al (2020) Blockchain based smart contracts for internet of medical things in e-healthcare. Electronics 9(10):1609
21. Shuaib K, Saleous H, Shuaib K, Zaki N (2019) Blockchains for secure digitized medicine. J Personal Med 9(3):35
22. Wang S, Yuan Y, Wang X, Li J, Qin R, Wang FY (2018) An overview of smart contract: architecture, applications, and future trends. In: 2018 IEEE intelligent vehicles symposium (IV). IEEE, pp 108–113
23. Witchey NJ (2019) Healthcare transaction validation via blockchain, systems and methods, 2 Jul 2019, US Patent 10,340,038
24. Zghaibeh M, Farooq U, Hasan NU, Baig I (2020) Shealth: a blockchain-based health system with smart contracts capabilities. IEEE Access 8:70030–70043
25. Zhang P, Walker MA, White J, Schmidt DC, Lenz G (2017) Metrics for assessing blockchain-based healthcare decentralized apps. In: 2017 IEEE 19th international conference on e-health networking, applications and services (Healthcom). IEEE, pp 1–4
26. Zhang R, George A, Kim J, Johnson V, Ramesh B (2019) Benefits of blockchain initiatives for value-based care: proposed framework. J Med Internet Res 21(9):e13595

Comparative Analysis of Breast and Prostate Cancer Prediction Using Machine Learning Techniques

Samta Rani, Tanvir Ahmad, and Sarfaraz Masood

Abstract Around the whole world, cancer is the most life-threatening disease. Basically, cancer can arise in any tissue of the body, and while each variety of cancer has unique characteristics, the fundamental processes that might cause cancer are highly common in all disease types. Breast cancer is one of the most ubiquitous types of cancer in females. In males, prostate cancer is the most dangerous during recent years. This study focuses on breast cancer as well as on prostate cancer in the direction of their early predictions. For early prediction, eight classification models had been used such as logistic regression (LR), Naïve Bayes (NB), decision tree (DT), stochastic gradient descent (SGD), K-nearest neighbors (KNN), decision tree (DT), random forest (RF), support vector machine (SVM), and artificial neural network (ANN). This work includes three different datasets for research analysis of breast and prostate cancer predictions. Two datasets for breast cancer (Coimbra and Wisconsin) and one for prostate cancer are taken from UCI and Kaggle repository, respectively. For improving the results of prediction, the normalization technique and feature selection method had been used in this paper. Performance in terms of accuracy, precision, recall, F1-score, and curves of each classifier are analyzed in this study. Most of the classifiers did well after using the feature selection method (ANOVA). In the case of Breast Cancer Coimbra, KNN give good results with 80% accuracy in both the cases with or without using feature selection. Logistic regression with feature selection doing the best work on Wisconsin Breast Cancer with 99% accuracy. There are four classifiers (SVM, RF, DT, and SGD) which gives highest accuracy (97%) on prostate cancer.

S. Rani (✉) · T. Ahmad · S. Masood
CSE, Jamia Millia Islamia University, Delhi, India
e-mail: rani.samta@gmail.com

T. Ahmad
e-mail: tahmad2@jmi.ac.in

S. Masood
e-mail: smasood@jmi.ac.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_54

Keywords Breast cancer · Prostate cancer · Feature selection · Normalization · Classifications

1 Introduction

Cancer is one of the leading causes of mortality worldwide, based on WHO statistics. Breast cancer is the second most common cancer, after lung cancer, with 2.09 million cases among the predicted 9.6 million cancer fatalities. It is also the fifth most prevalent cause of cancer death, accounting for over 627,000 fatalities, or 15% of all cancer deaths among women. And breast cancer alone accounts for 30% of all new cancer diagnoses in women [1]. This work examined the breast cancer issue using publicly available data from the Portuguese city of Coimbra and Wisconsin. There were ten quantitative predictor factors in this dataset, which were anthropometric in nature and captured through standard blood tests used to determine the presence or absence of breast cancer. Breast cancer is the most frequent type of cancer in women, affecting about 2.1 million women each year and contributing to female cancer deaths being the leading cause of death. Breast cancer claimed the lives of over 627,000 women in 2018. Early detection is crucial for improving breast cancer and survival chances [2]. Prostate cancer is one of the most frequent malignancies in American males, and it has the second highest fatality rate after lung cancer. Now a days, one in every seven men would be diagnosed with prostate cancer. According to recent figures, the number of new patients diagnosed with prostate cancer in 2017 was approximately 161,360, with approximately 26,730 deaths [3]. Fortunately, if prostate cancer is detected early, the mortality rate can be reduced. This paper also includes the study on prostate cancer whose dataset is taken from Kaggle and analyzes all classification models on parameters of prostate cancer. This paper is organized as follows: Sect. 1 presents the introduction to the different types of cancer disease. Section 2 presents the review of various recent literatures for cancer detection. Section 3 describes each component of the methodology used in this work, which is followed by description of the datasets. The results obtained after various experiments are presented and discussed in Sect. 5 followed by the conclusion.

2 Related Work

Rahman et al. [4], the purpose of this research is twofold. The first is to identify the most relevant breast cancer biomarkers, and second is to improve the current computer-aided diagnostic (CAD) system for detecting early breast cancer. This work made use of a dataset that included nine anthropometrical and clinical variables. From all the techniques used by author, SVM model with radial basis function (RBF) kernel gives best results with 93.9% accuracy, 95% sensitivity, and 94% specificity.

Ray et al. [5], in this study, researchers worked on two different datasets. One dataset is based on diabetic, and another is based on breast cancer. Feature selection techniques also applied before applying the machine learning models for getting the reduced feature set to classify between healthy and non-healthy subjects. Feature set includes the features having majority that is generated by routine pathology examinations. Author focused on identifying biomarkers that entail pathological testing and those that do not.

Mushtaq et al. [6], in this research, breast cancer (Wisconsin) dataset was used for study. Different classification models are applied along with PCA reduction approach. Performance of different classifiers with variants of PCAs based on linear, sigmoid, cosine, poly, and radial basis functions is analyzed. Highest 99.20% accuracy got from sigmoid-based Naive Bayes. Using KNN, with all different kernels got accuracies within the range 96.4–97.8%.

Shakeel et al. [7] works on prostate cancer for which author initially collects information related to prostate cancer from DBCR dataset. After that, using mean mode process, irrelevant record was removed and collect other important elements using ant rough set hypothesis. Result is evaluated in the terms of mean square error rate, hit rate, and accuracy.

3 Proposed Methodology

Figure 1 depicts the workflow of proposed work, highlighting the overall steps taken in this work, which includes data preprocessing with normalization, feature selection techniques, training and testing with specified models, evaluation of results, and

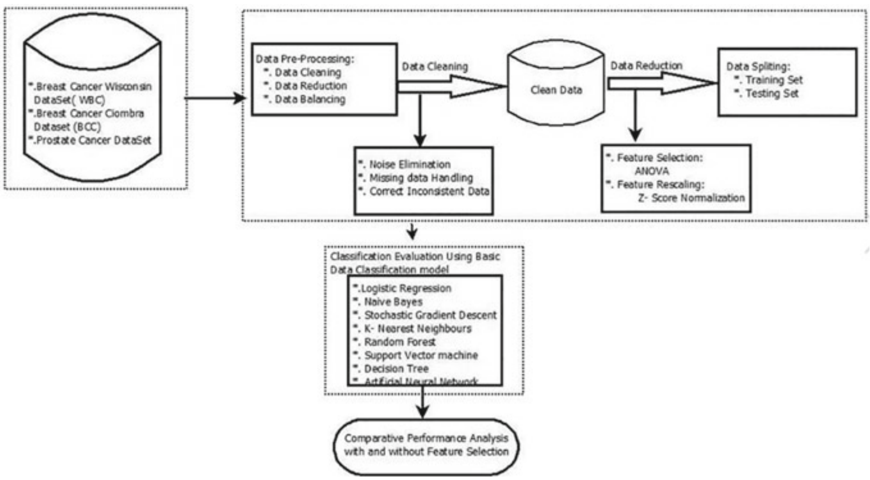


Fig. 1 Model for predicting cancer disease

Table 1 Description of breast and prostate cancer datasets

Dataset name	Total number of Patients	Number of parameters	Number of non-cancer patients	Number of cancer patients
Breast cancer coimbra data set (BCC)	116	10	52 (45%)	64 (55%)
Breast cancer wisconsin (diagnostic) data set (WBC)	569	32	357 (63%)	212 (37%)
Prostate cancer	100	10	38 (38%)	62 (62%)

prediction of breast cancer and prostate cancer. Python 3 was used to carry out this task.

Dataset

In this paper, three datasets had been used or analyzed for covering the famous cancer types in both males and females. Two datasets are based on breast cancer named as Breast Cancer Coimbra dataset and Breast Cancer Wisconsin, both had been collected from UCI repository. Third dataset had been collected from Kaggle named as prostate cancer. Table 1 shows the number of records under cancerous and non-cancerous cases in each dataset.

Coimbra Breast Cancer dataset has clinical parameters like body mass, hormone, leptin, glucosamine, etc. But another dataset which is also a breast cancer dataset WBC includes the real-valued parameters for each cell nucleus like texture, radius, compactness, etc. In this dataset, for each image, mean, standard error, and worst values were computed. Prostate cancer dataset having ten features like area, perimeter, radius, identification number, etc. In this paper, label 0 is used for non-cancer patients and label 1 for cancer patients.

4 Result Analysis

The proposed work considers eight classifiers for the analysis of performance comparison. Two normalization methods Z-score and min–max are used for data transformation. But, in this paper, only best results are discussed. Out of Z-score and min–max, Z-score gives good results. Tables 2, 3, and 4 show the results of BCC dataset, WBC dataset and prostate cancer dataset, respectively, using all the machine learning techniques. Every table divided into two parts having results based on without using ANOVA and with ANOVA.

Table 2 shows the comparison of results using the eight classifiers without feature selection and with feature selection on Breast Cancer Coimbra dataset. All classifiers

Table 2 Performance analysis of BCC

Models		Without feature selection				With feature selection			
		Precision	Recall	F1-score	Acc (%)	Precision	Recall	F1-score	Acc (%)
LR	0	0.64	0.41	0.50	60	0.67	0.71	0.69	69
	1	0.58	78	0.67		0.71	0.67	0.69	
KNN	0	0.86	0.71	0.77	80	0.75	0.88	0.81	80
	1	0.76	0.89	0.82		0.87	0.72	0.79	
NB	0	0.55	0.65	0.59	57	0.56	0.82	0.67	60
	1	0.60	0.50	0.55		0.70	0.39	0.50	
ANN	0	0.55	0.65	0.59	57	0.56	0.82	0.67	60
	1	0.60	0.50	0.55		0.70	0.39	0.50	
SVM	0	0.75	0.53	0.62	69	0.73	0.65	0.69	71
	1	0.65	0.83	0.73		0.70	0.78	0.74	
RF	0	0.75	0.53	0.62	69	0.73	0.65	0.69	71
	1	0.65	0.83	0.73		0.70	0.78	0.74	
DT	0	0.75	0.53	0.62	69	0.73	0.65	0.69	71
	1	0.65	0.83	0.73		0.70	0.78	0.74	
SGD	0	0.75	0.53	0.62	69	0.73	0.65	0.69	71
	1	0.65	0.83	0.73		0.70	0.78	0.74	

except the KNN give better results after using ANOVA. KNN classifier gives highest accuracy which is 80% and it remain same in both cases with or without feature selection.

Table 3 shows the performance of Wisconsin Breast Cancer dataset using all models. Logistic regression gives best result with 99% accuracy using ANOVA feature selection method. Here, only Naïve Bayes, logistic regression, and ANN classifiers improve their accuracies after using feature selection. Table4 showing the results of applied classifiers on prostate cancer dataset. Highest accuracy 97% is computed by five classifiers (NB, SVM, RF, DT, SGD). But the only difference is that Naïve Bayes gives best result without using feature selection and remaining classifiers gives their best accuracies after using ANOVA feature selection technique.

Figure 2 showing the learning curves of classifiers who gives highest accuracy in each dataset. In Fig. 2, curve (a) is showing the performance of KNN on Breast Cancer Coimbra dataset, curve (b) is showing the learning curve of logistic regression on Wisconsin Breast Cancer dataset, and curve (c) showing the results of support vector machine model on prostate cancer.

Table 3 Performance analysis of WBC

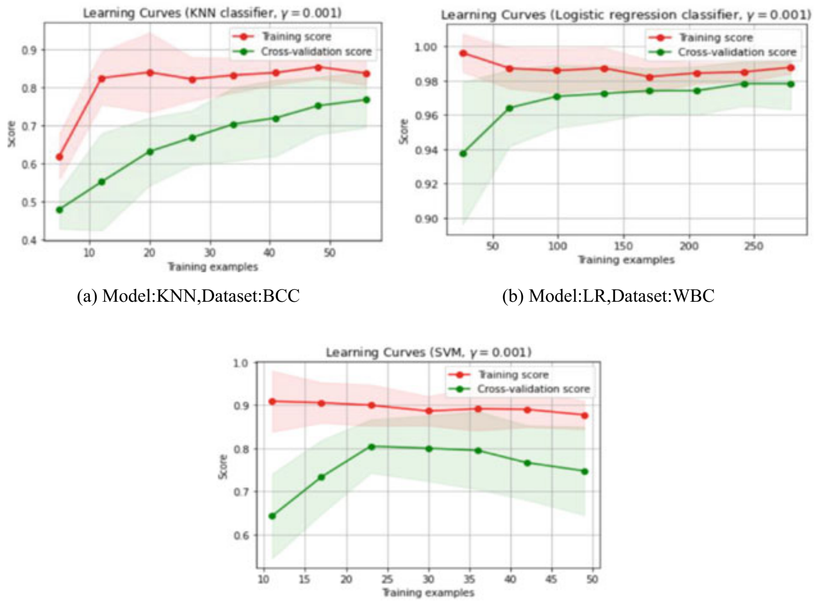
Models		Without feature selection				With feature selection			
		Precision	Recall	F1-score	Acc (%)	Precision	Recall	F1-score	Acc (%)
LR	0	0.97	0.99	0.98	98	0.99	0.99	0.99	99
	1	0.98	0.95	0.97		0.98	0.98	0.98	
KNN	0	0.95	0.99	0.97	96	0.96	0.96	0.96	0.95
	1	0.98	0.90	0.94		0.94	0.94	0.94	
NB	0	0.94	0.92	0.93	91	0.94	0.93	0.93	92
	1	0.86	0.90	0.88		0.88	0.90	0.89	
ANN	0	0.94	0.92	0.93	91	0.94	0.93	0.93	92
	1	0.86	0.90	0.88		0.88	0.90	0.89	
SVM	0	0.97	0.99	0.98	98	0.98	0.97	0.98	97
	1	0.98	0.95	0.97		0.95	0.97	0.96	
RF	0	0.97	0.99	0.98	98	0.98	0.97	0.98	97
	1	0.98	0.95	0.97		0.95	0.97	0.96	
DT	0	0.97	0.99	0.98	98	0.98	0.97	0.98	97
	1	0.98	0.95	0.97		0.95	0.97	0.96	
SGD	0	0.97	0.99	0.98	98	0.98	0.97	0.98	97
	1	0.98	0.95	0.97		0.95	0.97	0.96	

5 Conclusion

This work covers two main cancer types breast cancer (in females) and prostate cancer (in males) which are most dangerous and increase the mortality rate in whole world. It is very necessary to predict these diseases in their early stage for better treatment of patient. For early and correct predictions, all classification models are analyzed on each dataset. For improving the performance of models, firstly Z-score normalization method is used and analyze all the measuring parameters such as precision, recall, F1-score, and accuracy with or without using feature selection technique. The future anticipates the use of the aforementioned strategies to eliminate existing shortcomings and improve prediction rates, so giving a way to improve the survival rate for the well-being of mankind.

Table 4 Performance analysis of prostate cancer

Models		Without feature selection				With feature selection			
		Precision	Recall	F1-score	Acc (%)	Precision	Recall	F1-score	Acc (%)
LR	0	0.86	1.00	0.92	97	0.67	1.00	0.80	90
	1	1.00	0.96	0.98		1.00	0.89	0.93	
KNN	0	0.50	0.67	0.57	80	0.71	0.83	0.77	90
	1	0.91	0.83	0.87		0.96	0.92	0.94	
NB	0	0.45	0.83	0.59	77	0.45	0.83	0.59	77
	1	0.95	0.75	0.84		0.95	0.75	0.84	
ANN	0	0.45	0.83	0.59	77	0.45	0.83	0.59	77
	1	0.95	0.75	0.84		0.95	0.75	0.84	
SVM	0	0.57	0.67	0.62	83	1.00	0.83	0.91	97
	1	0.91	0.88	0.89		0.96	1.00	0.98	
RF	0	0.57	0.67	0.62	83	1.00	0.83	0.91	97
	1	0.91	0.88	0.89		0.96	1.00	0.98	
DT	0	0.57	0.67	0.62	83	1.00	0.83	0.91	97
	1	0.91	0.88	0.89		0.96	1.00	0.98	
SGD	0	0.57	0.67	0.62	83	1.00	0.83	0.91	97
	1	0.91	0.88	0.89		0.96	1.00	0.98	



(a) Model:KNN,Dataset:BCC

(b) Model:LR,Dataset:WBC

(c) Model: SVM, Dataset: Prostate Cancer

Fig. 2 Learning curves of models having best accuracy in every dataset

References

1. Prabadevi B, Deepa N, Krithika LB, Vani V (2020) Analysis of machine learning algorithms on cancer dataset. In: 2020 international conference on emerging trends in information technology and engineering (ic-ETITE), IEEE, pp 1–10
2. Asri H, Mousannif H, Al Moatassime H, Noel T (2016) Using machine learning algorithms for breast cancer risk prediction and diagnosis. Proc Comput Sci 83:1064–1069
3. Reda I, Ayinde BO, Elmogy M, Shalaby A, El-Melegy M, Abou El-Ghar M, Abou El-fetouh A, Ghazal M, El-Baz A (2018) A new CNN-based system for early diagnosis of prostate cancer. In: 2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018), IEEE, pp 207–210
4. Rahman MM, Ghasemi Y, Suley E, Zhou Y, Wang S, Rogers J (2021) Machine learning based computer aided diagnosis of breast cancer utilizing anthropometric and clinical features. Irbm 42(4):215–226
5. Ray A, Ray H (2021) Performance analysis of machine learning classifiers on different healthcare datasets. In: Emerging technologies in data mining and information security, Springer, Singapore, pp 99–111
6. Mushtaq Z, Yaqub A, Hassan A, Feng Su S (2019) Performance analysis of supervised classifiers using PCA based techniques on breast cancer. In: 2019 International conference on engineering and emerging technologies (ICEET), IEEE, pp 1–6
7. Shakeel PM, Manogaran G (2020) Prostate cancer classification from prostate biomedical data using ant rough set algorithm with radial trained extreme learning neural network. Health Technol 10(1):157–165
8. Smita EK (2021) Probabilistic decision support system using machine learning techniques : a case study of Cardiovascular diseases. J Disc Math Sci Cryptogr (JDMC) 1487–1496
9. Doja MN, Kaur I, Ahmad T (2020) Age-specific survival in prostate cancer using machine learning. Data Technol Appl
10. Masood S, Luthra T, Sundriyal H, Ahmed M (2017) Identification of diabetic retinopathy in eye images using transfer learning. In: 2017 International conference on computing, communication and automation (ICCCA), IEEE, pp 1183–1187
11. Kourou K, Exarchos TP, Exarchos KP, Karamouzis MV, Fotiadis DI (2015) Machine learning applications in cancer prognosis and prediction. Comput Struct Biotechnol J 13:8–17
12. Turgut S, Dağtekin M, Ensari T (2018) Microarray breast cancer data classification using machine learning methods. In: 2018 electric electronics, computer science, biomedical engineerings' meeting (EBBT), IEEE, pp 1–3
13. Gao K, Wang D, Huang Y (2018) Cross-cancer prediction: a novel machine learning approach to discover molecular targets for development of treatments for multiple cancers. Cancer Informat 17:1176935118805398
14. Polat K, Sentürk U (2018) A novel ML approach to prediction of breast cancer: combining of mad normalization, KMC based feature weighting and AdaBoostM1 classifier. In: 2018 2nd International symposium on multidisciplinary studies and innovative technologies (ISMSIT), IEEE, pp 1–4
15. Mehdi M, Pahwa K, Sharma B (2019) Comparison of data mining algorithms for predicting the cancer disease using python. In: 2019 8th International conference system modeling and advancement in research trends (SMART), IEEE, pp 155–160

Data-Driven Volatile Cryptocurrency Price Forecasting via Variational Mode Decomposition and BiLSTM

Rohith Ramakrishnan, Anirudh Vadakedath, Anirudh Bhaskar, S. Sachin Kumar, and K. P. Soman

Abstract Cryptocurrency is based on blockchain technology which is ideally decentralised, referring to no superior authority overlooking it. The community is maintained by numerous user machines forming a "peer-to-peer" network. With the recent skyrocket of crypto-assets in the financial markets, many view it as the quickest and riskiest way to earn. Such assets are coined as "Volatile" due to its rapidly fluctuating price, thereby making it extremely hard to forecast its course. The paper at hand explores a novel technique that establishes a relation between signal processing and volatile stock forecasting methods via variational mode decomposition (VMD). Variational mode decomposition aided with BiLSTM neural architecture, a purely data-driven model, is fine-tuned to forecast the daily or interday prices of Bitcoin and Ethereum alongside yielded RMSE of 0.0278 for Bitcoin. The results are then further compared with ARIMA, ARMA and MA to analyse the effect of VMD.

Keywords Variational mode decomposition · VMD · Signal processing · Cryptocurrency · Bitcoin · Etherium · Price forecast · BiLSTM · Volatile stocks · Time series

1 Introduction

Numerous researches over the years have been articulated to develop efficient models for forecasting the prices of volatile assets such as Bitcoin(BTC) using several different data-driven approaches. In statistics, volatility refers to the amount of dispersion

R. Ramakrishnan · A. Vadakedath · A. Bhaskar · S. Sachin Kumar (✉) · K. P. Soman
Centre for Computational Engineering and Networking, Amrita School of Engineering Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, India
e-mail: s_sachinkumar@cb.amrita.edu

R. Ramakrishnan
e-mail: cb.en.u4aie19052@cb.students.amrita.edu

A. Vadakedath
e-mail: cb.en.u4aie19008@cb.students.amrita.edu

A. Bhaskar
e-mail: cb.en.u4aie19007@cb.students.amrita.edu

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_55

of profit for any index. In most cases, the higher the volatility, the riskier the security. It is often measured as either the standard deviation or the variance between returns from that same security or market index [1]. One of the major hurdles in developing a forecasting model is the volatile nature of the assets. Volatility has always been seen as a system that can never be predicted. The upcoming systems such as cryptocurrency, which have the potential to become institutions for future markets, get hampered by its high volatility. Volatility has been a huge hurdle due to its nonlinear nature; i.e. the output is not proportional to the input. The idea of understanding the paths of nonlinear systems had always been a dream until an approach to comprehend systems using data-driven approaches came to light. Traders around the world run millions of computations which results in many important decisions. Being able to understand the market sentiment and predict the rise and fall of shares has been crucial. The recent stardom of cryptocurrencies has drawn the attention of numerous investors due to its rapid inclination in the past few months alone. Despite its ban in numerous countries [2], it continues to astonish investors. The cryptocurrency was originally developed as a decentralised alternative to the existing banking and currency system. Cryptocurrency evolved with advancements in blockchain technology. Blockchain and distributed ledgers are rapidly emerging and provoking various sectors in different fields. One such industry is the financial industry, which is utilising blockchain to revolutionise payment methods [3]. This impact can be illustrated by the existing payment process. Products paid by credit card are completely settled after numerous business days. The evolving blockchain can eliminate delays and make these settlements anonymous [4]. Such captivating features lead to "mooning" or skyrocketing prices of numerous cryptocurrencies which are now seen as digital gold. With the scope of attaining easy profits, numerous researchers and people are looking for ways to predict and forecast the prices of volatile assets to minimise loss. In this paper, an attempt to make a prognosis or forecast of the inter-day prices of two reputed cryptocurrencies, Bitcoin (BTC) and Ethereum (ETH), is contemplated as signals and decomposed into modes by the application of variational mode decomposition (VMD) on each of them. Decomposition into modes will help assist in capturing the volatility and irregularities in the price of each asset and better train of BiLSTM model to attain finer predictions. Section 2 in this paper throws light on the existing approaches for forecasting prices of numerous assets. Sections 3 and 4 give a basic understanding on the theory and the model. Section 5 provides information on preprocessing of data used along with the metrics used to evaluate the model and the values obtained followed by conclusion.

2 Literature Review

Volatile markets are a huge risk for traders, and predictive algorithms are of substantial help in predicting stock prices. LSTM neural networks and recurrent reinforcement learning were used by Lu [5, 6] to maximise an objective function through gradient ascent. An agent was chosen, and based on the function value, it would decide how it would trade. LSTM was implemented here to sense the market conditions dynamically since in theory RNN's can connect previous information to the present task. Natural gas prices are another market where artificial neural networks were used to predict short-term prices [7, 8]. Jin et al. [7] decomposed the spot gas price using discrete wavelet decomposition and then autoregressive integrated moving average (ARIMA) followed by ANN are applied to this for forecasting. Dunis et al. [9] used neural networks to check whether it could predict the EUR/USD exchange rates during the financial crisis of 2007–2009. The Foreign Exchange market is highly volatile, and a sample period from January 2000 to July 2007 was provided to the neural network to learn from. Kimoto et al. [10] used a data-driven approach and used multiple factors such as interest rates, vector curves, foreign exchange rates, New York Dow Jones average, and turnover to predict the selling price. İcan et al. [11, 12] mentions how ANN combined with another ML model can yield better results. Chen et al. [13] used ten-year data ranging from January 1982 to August 1992, using the first 5 years to train the model and the next 5 years to evaluate it. This study uses a probabilistic neural network (PNN), generalised method of moments (GMM) with a Kalman filter, and a random walk model to predict the direction of future index returns. Bernal, et al. [14] compared using echo state network (ESN) and a Kalman filter to predict a time series. The study used data ranging from 2004 to 2009 to conclude that while an ESN could capture quick changes in stock price, the Kalman filter could not. Silpa Balagopal et al. had applied multivariate variational mode decomposition to forecast prices of numerous stock sectors [15]. Hiransha M et al. proposed a comparison of multiple deep learning architectures naming a few, multilayer perceptron (MLP), recurrent neural networks (RNN), long short-term memory (LSTM) and convolutional neural network(CNN) [16, 17]. The stock prices of five different stocks on the National Stock Exchange (NSE) and New York Stock Exchange (NYSE) were predicted using these networks, given the price of one company on the NSE. These results were compared with linear models such as ARIMA, autoregressive moving average (ARMA), autoregression (AR) and moving average (MA). Deepthi P K et al. proposed a method to predict stock prices [18] and Dr. Neethu proposed electric load forecasting [19] using dynamical mode decomposition (DMD). Each company's price was predicted in three ways. Firstly, the organisations belonging to the identical zones were sampled by which the future price was predicted, while in the other technique, they took into consideration sampled organisations from all sectors. The sampling and the prediction window size were kept fixed in all these cases. In the third approach, sampling was carried out by taking into consideration organisations from all sectors. Here the window used for sampling was kept constant; however, until a determined cut-off error was crossed, the predictions were made.

3 Variational Mode Decomposition

The objective of the novel VMD proposed by Konstantin and Zosso [20] is the successor to the empirical mode decomposition (EMD) and aims at decomposing

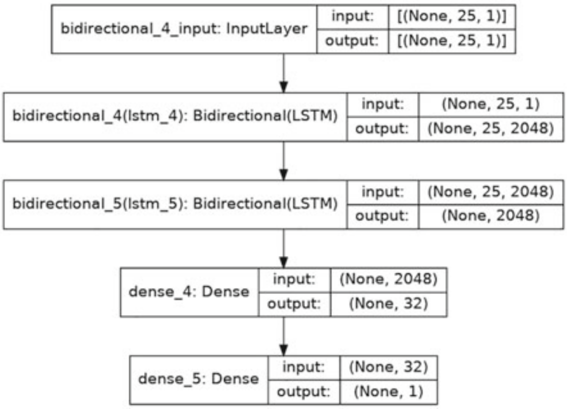


Fig. 1 Architecture of the proposed BiLSTM

any signal f(t) to a cumulative sum of functions, uniquely called, intrinsic mode functions [21]. Each mode k must be largely compact around a central pulse k that is identified throughout the decomposition process [20]. Consider f as a time series that, according to the restricted optimisation problem, is decomposed into a set of modes u_k centred around pulsation w_k.

$$\min_{\{u_k\},\{\omega_k\}} \left\{ \sum_k \left\| \partial_t \left[\left(\delta(t) + \frac{j}{\pi t} \right) * u_k(t) \right] e^{-j\omega_k t} \right\|_2^2 \right\} \quad \text{s.t. } \sum_k u_k = f \tag{1}$$

where $\{u_k\} := \{u_1, .., u_k\}$ and $\{w_k\} := \{w_1, .., w_k\}$ are the symbolic representation for the set of all modes and their associated central frequencies. In this study, the radiography data obtained from each channel will be considered as a collection of signals and hence decomposed into modes. The number of modes, as the name suggests decomposes the given signal into the desired number of modes (K) and α is the balancing parameter of the data-fidelity constraint.

4 BiLSTM Model

BiLSTM which stands for bidirectional long short-term memory learns long-term dependencies which are bidirectional between any given data sequence. The proposed architecture consists of BiLSTMs' and dense layers as shown in Fig. 1. The architecture was developed to be elementary to showcase the capability of VMD aided prediction on uncomplicated neural networks.

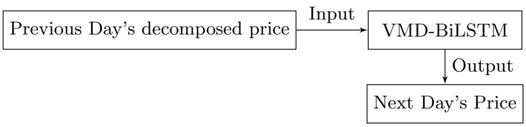


Fig. 2 High-level overview of the proposed system

5 Proposed Methodology

In the present section, the proposed model applied in the experiments will be elaborated. Figure 2 shows a high-level overview of the system. The system accepts the decomposed prices of the previous day, which assists the model in assessing the closing price of the selected assets. The model tries to learn from the previous days decomposed price signal to forecast the next day's price.

5.1 Data Preparation

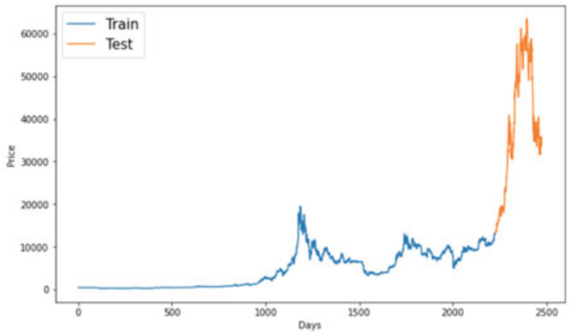
The dataset utilised for the experiments in this paper consists of BTC and ETH prices in US dollars (US$). The dataset is inclusive of 2638 observations from 11 April 2014 to 17 June 2021. The dataset includes five attributes closing, highest, lowest and opening price and volume. The scope of this experiment is restricted to the analysis and prediction of closing prices. The closing price for both crypto-assets was broken down into training and testing with the proportion of 9:1, respectively. In Fig. 3, the blue line indicates the data taken for training, whereas the orange indicates the testing. **Step-1**: The decomposed modes can be seen from Fig. 4. The first mode captures the trend of the signal.

Step-2: The length of each mode will be identical to the length of the closing price, and hence, each day's closing price is decomposed into 25 values by considering a window of length 2048

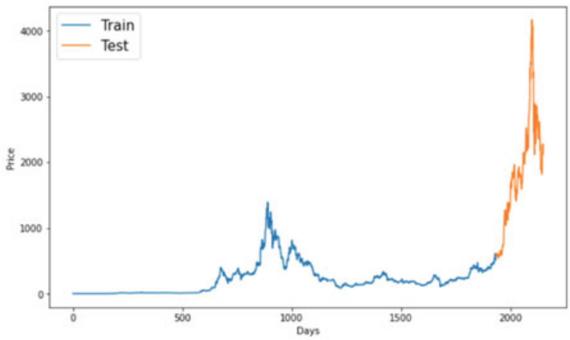
Step-3: The decomposed data will be rearranged in such a way that the previous day's decomposed value will point to the current day's price. Hence, the proposed model learns to forecast closing price based on the previous day's decomposed values.

Step-4: The realigned modes will act as the features which will be learned by the BiLSTM neural net and will forecast the closing price for the upcoming days. In the current work, BiLSTM net is only used as it captures the features from left-to-right and right-to-left manner.

Fig. 3 Train–test split



(a) Bitcoin



(b) Ethereum

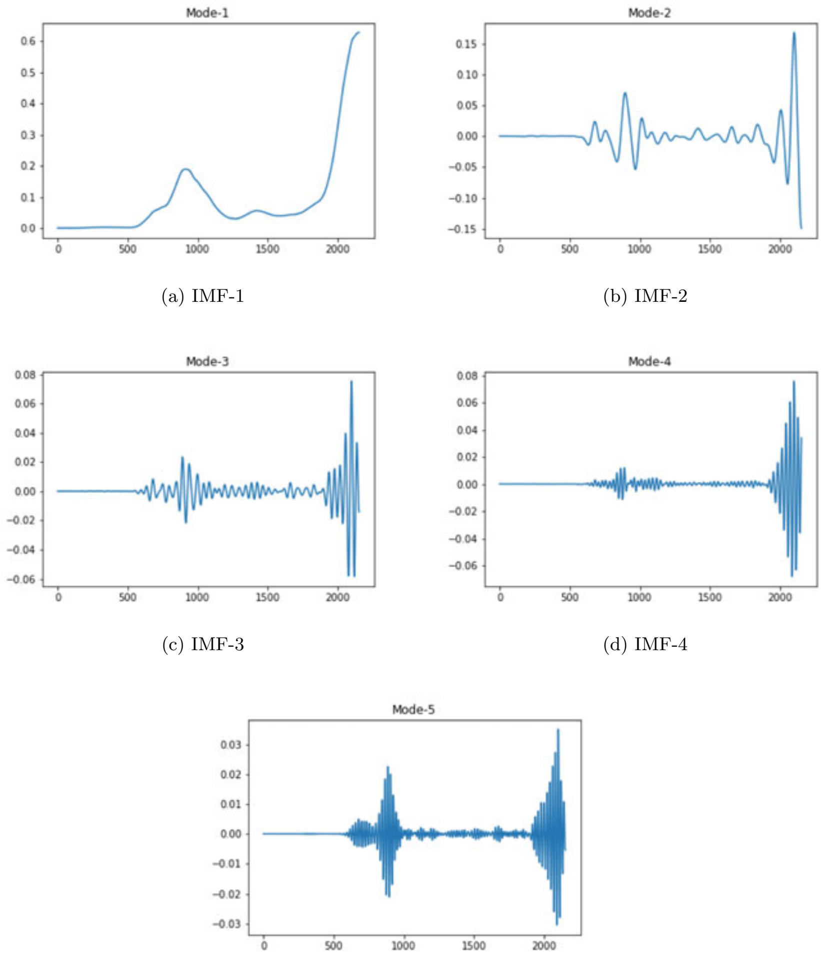
5.2 *Performance Indicators*

Two indicators were used to evaluate the model's performance: root mean square error (2) (RMSE) and mean absolute error (3) (MAE), both of which can be calculated and described as

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{\text{Pred}_i \text{ - Actual}_i}{\sigma_i}\right)} \tag{2}$$

$$\text{mae} = \left(\frac{1}{n}\right)\sum_{i=1}^{n}|\text{Pred}_i - \text{Actual}_i| \tag{3}$$

in Eqs. (2) and (3), n denotes the length of the observation sequence.



(a) IMF-1

(b) IMF-2

(c) IMF-3

(d) IMF-4

(e) IMF-5

Fig. 4 Top 5 IMFs obtained via VMD for closing price of BTC

6 Results and Discussion

The first mode, which is obtained via decomposing the BTC closing price, will capture the general trend of increase in the past couple of months, while rest of the 24 modes capture the "volatile" or variations in the price in the price as seen in Fig. 5.

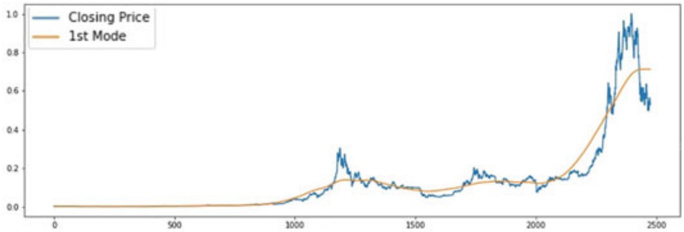
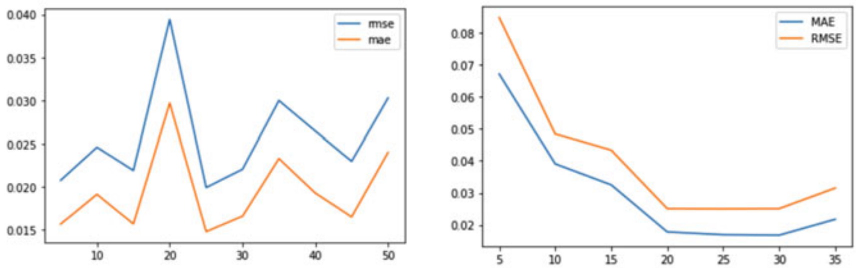


Fig. 5 First mode of closing price which captures the general trend



(a) Bitcoin (b) Ethereum

Fig. 6 Hyper-tuning K values for BTC and ETH in obtaining least RMSE and MAE

6.1 Hyper-Tuning K Value

One of the essential parameters for VMD is the K, the number of modes the given signal is decomposed into. The above-mentioned procedures are evaluated with the performance metrics for numerous K values as depicted in Fig. 6a for BTC and Fig. 6b for ETH.

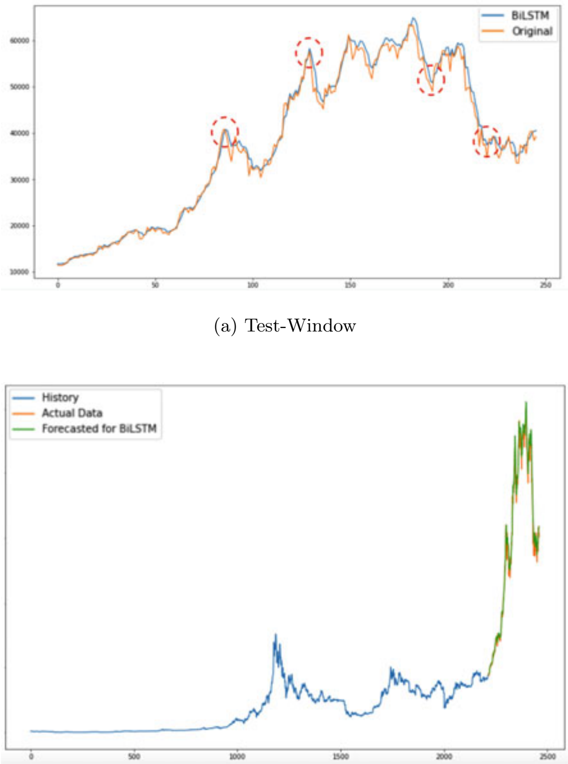
6.2 Forecasted Price

After hyper-tuning, the proposed methodology was applied for closing prices of BTC and ETH. One of the key issues in predicting BTC prices or time series is the random fluctuation or volatility in certain time periods, to evaluate the models performance during such months, July 2020 and July 2021 have been chosen.

From Figs. 7 and 8, it can be observed that the proposed approach is able to forecast matching trend of the original price. The predicted trend is matched with the original price. The captured trend consists of some spikes and dips which are marked in dotted circles.

From Fig. 9, its evident the turning points and trends of BTC closing prices are captured progressively better.

Fig. 7 Comparison of forecasted versus actual BTC price



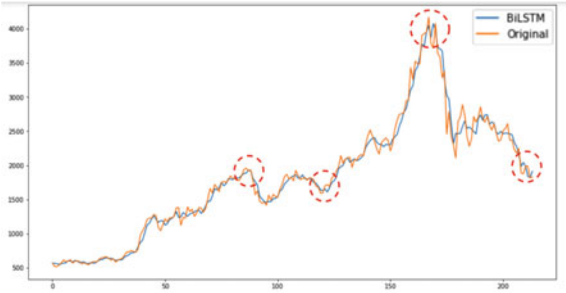
(a) Test-Window

(b) History

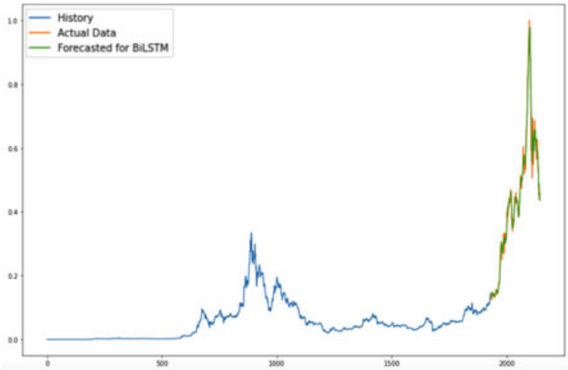
6.3 Correlating with Existing Methods

According to the« literature, forecasting may be done using two approaches: statistical and artificial intelligence techniques [22]. It is commonly used in the disciplines of finance and economics [23]. In this section, we will be comparing the performance of the proposed VMD-BiLSTM methodology with the existing ARIMA-based prediction. The data was split as mentioned in Sect. 5.1, and its prediction was noted.

To evaluate the performance of VMD-based forecasting, ARIMA [24], ARMA and MA, the above-mentioned performance metrics were used. The same time frame was used to forecast prices with methods such as ARIMA, ARMA, and MA and its performance metrics were compared to those of our proposed VMD-BiLSTM method and tabulated in Table 1.

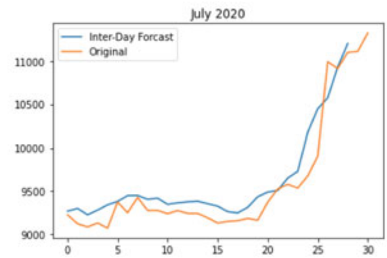


(a) Test-Window

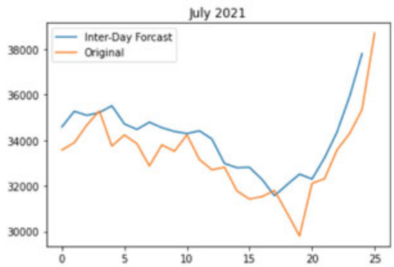


(b) History

Fig. 8 Comparison of forecasted versus actual ETH price



(a) July 2020



(b) July 2021

Fig. 9 Comparison of forecasted versus actual BTC price for the months of July

Fig. 10 Prediction of closing price via ARIMA

Table 1 Comparing proposed approach with ARIMA, MA, AMRA for BTC

Method	MAE	RMSE
VMD-BiLSTM	0.0207	0.0278
VMD-LSTM	0.5586	0.6051
ARIMA	0.8079	0.8920
MA	0.5107	0.5107
ARMA	0.4287	0.4926

Figure 10 corresponds to the prediction of BTC via ARIMA model which was trained and tested on the same windows as VMD-BiLSTM model. Muniye et al. have proposed the LSTM-based prediction model for BTC [25] with RMSE value of 0.092, our proposed model aided with VMD obtained a RMSE score of 0.027 .

6.4 Limitations

Due to the extensive list of parameters for VMD, each of them must be hyper-tuned for the finest results. The experiments performed in this paper were iteratively hyper-tuned, which was computationally expensive.

7 Conclusion

Volatile time series data such as cryptocurrency prices and its forecasting is capturing the attention nowadays. The present paper proposes a data-driven approach using VMD and BiLSTM for forecasting the closing price of BTC and ETH. VMD is a promising method which captures the trend and variations in its mode and the previous day mode values are chosen as feature vectors for the BiLSTM net. The results are compared with ARIMA, ARMA and MA. From the results obtained via the proposed approach, it can be observed that it is able to capture some of the trends in the forecast. As future work, it is planned to perform and extensive studies including the price data of other cryptocurrencies and deep neural nets.

References

1. Hayes A (2021) Volatility
2. Bajpai P (2021) Countries where bitcoin is legal and illegal
3. Michael N, Peter G, Oliver H, Dirk S (2017) Blockchain. Bus Inf. Syst Eng 59:03
4. Iansiti M, Lakhani KR. The truth about blockchain
5. Chen Z, Li C, Sun W (2020) Bitcoin price prediction using machine learning: an approach to sample dimension engineering. J Comput Appl Math 365:112395
6. Lu DW (2017) Agent inspired trading using recurrent reinforcement learning and lstm neural networks
7. Jin J, Kim J (2015) Forecasting natural gas prices using wavelets, time series, and artificial neural networks. Plos One 10(11):1–23
8. Selvin S, Vinayakumar R, Gopalakrishnan EA, Menon VK, Soman KP (2017) Stock price prediction using LSTM, RNN and CNN-sliding window model. In: 2017 international conference on advances in computing, communications and informatics (ICACCI), pp 1643–1647
9. Tripathi M, Kumar S, Inani SK (2021) Exchange rate forecasting using ensemble modeling for better policy implications. J Time Ser Econ 13(1):43–71
10. Kimoto T, Asakawa K, Yoda M, Takeoka M (1990) Stock market prediction system with modular neural networks. IJCNN Int Joint Conf Neural Netw 1:1–6
11. Akyildirim E, Goncu A, Sensoy A (2021) Prediction of cryptocurrency returns using machine learning. Ann Oper Res 297(1):3–36
12. İcan Ö, Çelik T (2017) Stock market prediction performance of neural networks: a literature review. Int J Econ Fin 9:100
13. Omer S, Murat O (2020) Financial trading model with stock bar chart image time series with deep convolutional neural networks. Intell Autom Soft Comput 26:06
14. Bernal A, Fok S, Pidaparthi R (2012) Financial market time series prediction with recurrent neural networks. Citeseer, State College
15. Balagopal S, Menon VK, Gopalakrishnan EA, Soman KP (2021) Multivariate variational mode decomposition based analysis on stock sectors. In: Paprzycki M, Thampi SM, Mitra S, Trajkovic L, El-Alfy EM (eds) Intelligent systems, technologies and applications. Springer, Singapore, pp 389–401
16. Hiransha M, Gopalakrishnan EA, Menon VK, Soman KP (2018) NSE stock market prediction using deep-learning models. Proc Comput Sci 132:1351–1362
17. Jay P, Kalariya V, Parmar P, Tanwar S, Kumar N, Alazab M (2020) Stochastic neural networks for cryptocurrency price prediction. IEEE Access 8:82804–82818
18. Kuttichira DP, Gopalakrishnan EA, Menon VK, Soman KP (2017) Stock price prediction using dynamic mode decomposition. In: 2017 international conference on advances in computing, communications and informatics (ICACCI). IEEE, pp 55–60
19. Neethu Mohan S, Soman KP, Sachin Kumar S (2018) A data-driven strategy for short-term electric load forecasting using dynamic mode decomposition model. J Appl Energy 20:229–244
20. Dragomiretskiy K, Zosso D (2013) Variational mode decomposition. IEEE Trans Signal Process 62(3):531–544

21. Zeiler A, Faltermeier R, Keck I, Tomé A, Puntonet C, Lang E (2010) Empirical mode decomposition—an introduction, pp 1–8
22. Wang J-J, Wang J-Z, Zhang Z-G, Guo S-P (2012) Stock index forecasting based on a hybrid model. Omega 40(6):758–766
23. Ariyo AA, Adewumi AO, Ayo CK (2014) Stock price prediction using the arima model. In: 2014 UKSim-AMSS 16th international conference on computer modelling and simulation, pp 106–112
24. Smith TG et al (2017) PMDARIMA: arima estimators for python [online, accessed today]
25. Muniye T, Rout M, Mohanty L, Satapathy S (2020) Bitcoin price prediction and analysis using deep learning models, pp 631–640

Artificial Intelligence Techniques to Restrain Fake Information

Lakshmi Narasimha Gunturu, Kalpana Pamayyagari, Girirajasekhar Dornadula, and Raghavendra Naveen Nimbagal

Abstract In the current world, there has been an upsurge in the use of social networking sites like Facebook, WhatsApp, Twitter, etc. These are considered suitable sites for the exchange of messages and sharing pictures and videos. Besides providing entertainment to the users, sometimes the information circulating on these platforms may be fake or misleading. In this chapter, we reviewed the literature on AI technologies that address the issue of fake news detection, the process of information flow, different data sets to detect fake news, and future perspectives to improve the credibility of information.

Keywords Artificial intelligence · Deep learning · Fake information · Future directions · Global risk · Social media

1 Introduction

Formerly news reports were developed and disseminated through traditional media like radio, newspapers, magazines, television, etc. Nevertheless, in the current era of the Internet, we have many networks that enable the generation and ease the expansion of news through social media. Online social media, like Twitter, Facebook and Instagram, provide a major podium for social interactivity and information transference far and wide among the users [1]. The breakthrough of technology allows everyone to build, obtain and unroll news messages rapidly and ubiquitously. Thus, it is difficult to recognize the genuineness of the open-out information. False information has a remarkable impact on individual value [2, 3]. The current chapter focus on a discussion of data sets that are involved in the detection of fake news and future perspectives to enhance the credibility of information.

L. N. Gunturu (✉) · K. Pamayyagari · G. Dornadula
Department of Pharmacy Practice, Annamacharya College of Pharmacy, Rajampeta, India
e-mail: gunturunarasimha007@gmail.com

R. Naveen Nimbagal
Department of Pharmaceutics, Sri Adichunchanagiri College of Pharmacy, B.G Nagara, Karnataka, India

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_56

2 Methodology

An extensive literature survey was performed in databases such as PubMed and Google Scholar by using keywords such as fake news, artificial intelligence, combat technologies and social media. All the related articles that fall under the scope of technology applications to detect fake news are taken into consideration.

3 Process of Information Flow

The customary mass media and journalism are accountable for content creation and its distribution. However, a prominent shift has occurred through social media where various parties have been involved in creation and dissemination (Fig. 1). In creation, an individual or a group can generate fake news for his or her interests or other third parties. The creation and spread of fake information are persuaded by various social, political, financial and malicious factors [4]. Dissemination is the social communicating tool meant for the intentional spread of information which is accomplished through a conventional methods involving provider and consumer [5]. Through digital platforms, various social sites such as Twitter, YouTube and Facebook had established for the propagation of information, collaboration and socialization. Dissemination of trending news, government policies, political debates, product reviews and personal and professional pursuits is feasible through the Internet community. Thus social networking systems turned out as a podium for the user population starting from usual chattering to the spread of newsbreak [6].

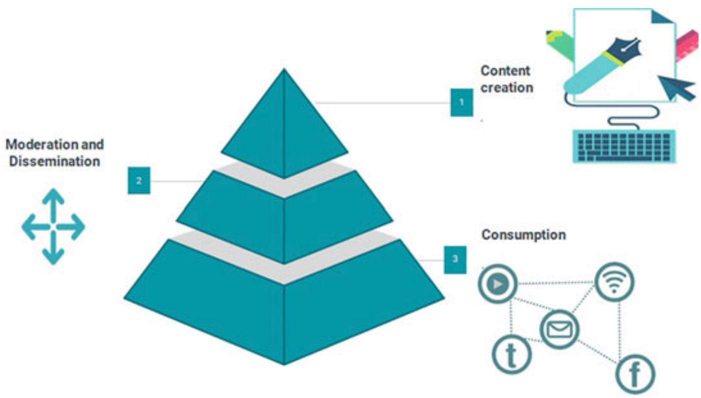


Fig. 1 Process of information flow

4 Results and Discussion

Our search identified the following potential algorithms that are reported in the literature to control the spread of fake news.

4.1 Scientific Content Analysis

The earliest works on assessing the linguistic cues were done by Driscoll. He examined the written statements of criminal investigators to obtain the information credibility by use of a new method named scientific content analysis (SCAN) [7]. It consists of cues concerned with false information detection. This clue includes the information content, structure and connections within the paragraphs. It also assesses the memory of an individual, missed information and use of emotional sentences. Initial examination results of SCAN by Driscoll reported the better results. However, further studies reported many drawbacks in this analysis of false information by SCAN methods [8]. The major limitation of this approach was it lacks the supportive evidence to prove its originality in terms of fake news detection. It also requires the need for highly skilled professionals to validate the results for originality.

4.2 Linguistic Cue Set

This was proposed by Fuller et al. in an attempt to minimize the involvement of humans in the detection of fake news by the use of an automated text method. He integrated many linguistic cue sets with previous existing cue sets [9]. Zhou et al. cue set were the first one among those, which comprises nearly 14 linguistic cues for the detection of false information. It included the sensory ratio, spatial–temporal ration, imaginary ratio and average word length [10]. The second set of cues had fitted with deceptive features taken from the deceptive theories. Those deceptive features in this set are word quantity, certainty terms and verb quantity [11, 12]. The third set of linguistic cues consists of 31 sets of cues that are created by combining first and second sets with an additional feature of linguistic inquiry word count. It also possesses lexical diversity passive verbs and modern verbs [13]. To estimate the importance of cue, Fuller et al. used three different classifiers like logistic regression, neural network and decision trees that provided better output results. One of the major limitations of this approach is the lack of generalizability in concepts of language, topics and domains. Ali and Levine et al. concluded that linguistic cue sets developed for one situation cannot be used as such as for another situation [14]. For example, a cue set used for accounting purposes cannot be used for the police enquiry. In further days, handcrafted cue works are proposed for the detection of fake news. In this process, Rubin et al. analysed several sentences with punctuation marks and sentiment of the text [15]. Zhao et al. later identified different expressions to identify the fake news circulating in social media with hashtags and posts. Later several linguistic cues are developed are social media platforms such as Twitter and Wikipedia [16]. The major limitation for this is a comprehensive listing of regular patterns of hashtags and sentences requires more effort.

4.3 N-gram Approach

This was the most popular approach used in the identification of fake news by linguistic methods [17, 18]. It consists of "n" adjoining sentences within a text, accountable for consisting of unigrams, bigrams and trigrams that are more frequently used in the text analysis. Mihalcea et al. [17] developed the n-gram for lie detection. Data sets for this are constructed based on crowdsourcing and the statements include the people's beliefs on abortions, the death penalty and personal emotions on family or friendship. In these conditions, they wanted to estimate the difference in texts and the accuracy of n-grams towards the detection of truth. Support vector machines and Naïve Bayes algorithms were used in this approach as input models. Their result reported it had an accuracy of 70% in reporting the people's beliefs and 75% accuracy in estimating the emotions of subjects. The limiting factor is, this approach is very simplified and using n-grams alone cannot able to detect the presence of false information.

4.4 Part Of Speech Tag

Apart from the use of n-gram features, part of speech hashtags are also used in the detection of fake news. They work by adjoining each word in a sentence according to their grammatical relations such as nouns, adjectives, etc. They vary according to conditions such as medical meetings, consultations and ceremonies [19]. Ott et al. in their study analysed the relation among variations of part of speech in text and truth veracity. He obtained a better result with n-gram techniques. Despite this, he found that part of speech tags is sound approach to judge human behaviours [20]. When this approach is used alone, it only provides the grammatical corrections and is weaker in contrast to word-based approaches like writing styles, which is considered as a major limiting factor.

4.5 Probabilistic Context-Free Grammar

Further works take into account the deeper features extracted from the probabilistic context-free grammar tress. This tree has terminal nodes and intermittent nodes.

Terminal nodes represent words and intermittent nodes denote the syntactic features [21]. Feng et al. used this approach by analysing the deeper features of the tree towards truth detection. During this approach, he proposed four variants as four production rules. The first variant does not possess any terminal nodes and has only production rules. The second variant consists of all the rules of products extracted from the given data set. In the third and fourth variant, there are small modifications to fit the grandparent nodes [22]. In this particular approach, classifier weights along with syntactic corrections are used in the detection of fake news from true news. The major limiting factor for this approach is it can able to identify only syntactic corrections in a sentence but is weaker in providing truth regarding sentence-sensitive information. Due to this reason, its use was limited to identification fake news in longer text or articles.

4.6 Deep Learning Methods

Deep learning is also called deep machine learning. It utilizes a group of algorithms whose function is similar to the human brain. It is a part of the machine learning programme. It has two neurons one to receive the input signal and the other to give the output. Deep learning methods are used in different areas such as language processing, speech recognition and fake news identification. This network is considered a good choice for truth identification in social media and articles. It can extract and assess information around the world at an enormous rate. Hence, this method is considered a good tool in the detection of fake news [23]. In this section, we present an overview of convolutional neural networks (CNN) for the detection of fake news. CNN mainly works in two phases one is called feature representations and the other was classification layers as shown in Fig. 2. The first phase of CNN transforms the targeted information into feature vectors. This vectors developed from the first phase are used as input by classification layers. CNN models adapt the unique style of creating a feature for the given input. Convolutional filters are used to identify the distance between adjacent words. Maxpooling layer avoids overfitting of information by forming a smaller dimension of information. Here are a few studies that utilized algorithms for the detection of fake news. Tacchini et al. concluded that the use of Boolean label crowd (BLC) setting and logistic regression reported an accuracy of 99% for logistic regression and 99.4% for BLC for fake news detection [24]. Pratiwiet et al. concluded that Naïve Bayes classification reported accuracy of nearly 78% in fake news detection [25]. Kim et al. deployed a curb algorithm that can effectively identify fake news [26].

Despite better results, identification of fake news remains a big challenge because the content is designed in a manner to resemble the truth and to cheat the readers. In addition, a lack of awareness on fact-checking makes the users believe the information quickly.

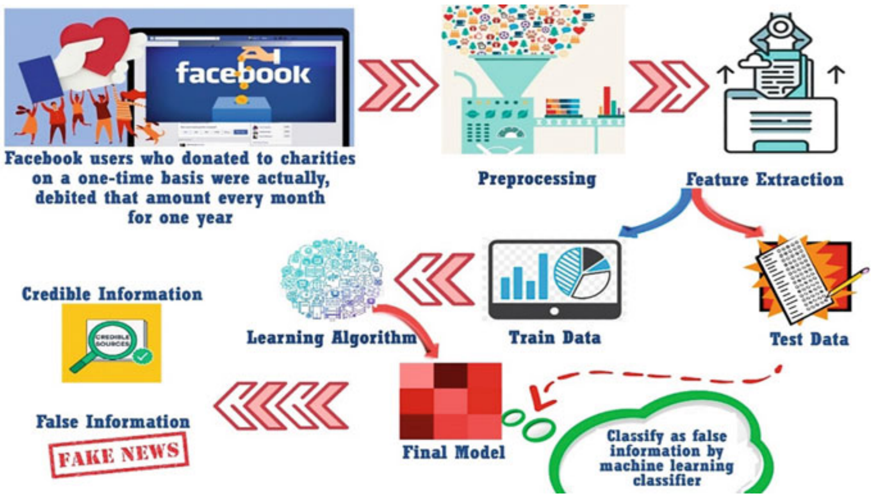


Fig. 2 Information flow by deep learning model

4.7 *Machine Learning Methods*

Machine learning is a discipline of artificial intelligence that provides the system ability to innately learn and enhance from the experience without being explicated. Examples of machine learning include supervised machine learning, unsupervised machine learning and semi-supervised algorithms. Supervised algorithms use labelled information to provide the input, unsupervised machine learning utilizes unlabelled information for the classification of data and semi-supervised machine learning is a mixture of both labelled and unlabelled data which falls in between the supervised and unsupervised algorithm [27]. This section deals with the process involved in the conversion of fake news into credible information by using machine learning algorithms. This generally involves stages like pre-processing, feature extraction, train test split and classifier application. Pre-processing involves in minimization of noise in the text by execution of lemmatization, stemming and removal of stop words to improve the classifier performance. In the next stage of feature extraction, relevant and handy data is obtained through online sources and used as an input for the next stage. In train text, split stage complete data is categorized into two sets of information as training and testing. Training data is used for learning algorithms; whereas, testing information is used to enhance the model performance. In the final stage of machine learning features belonging to similar classes are estimated as fake information or real information by using the trained data set [28]. The key differences between machine learning and deep learning algorithms are machine learning separates information into training data and uses a combination of different models to detect fake news but in deep learning algorithms, different layers are present which

Table 1 Machine learning algorithms for fake news detection

Study	Objective	Methodology	Result
Ma et al. [29]	Rumour detection in microblogs	Two text representative models and a Neuronal network model	It reported two text representative model has the accuracy of 96% in the detection of rumours when compared to the neuronal network model
Hamidian et al. [30]	Rumour detection and rumour classification	Single step RDC (SRDC) and two step RDC (TRDC)	TRDC has attained an F measure of 82.9% when compared to SRDC
Ahmad et al. [31]	Online fake news detection	N-grams and machine learning model	Accuracy of 92% was obtained by term frequency inverse document frequency

can analyse information differently. Summary of machine learning algorithms used in real work is mentioned in Table 1.

A study by Ball et al. concluded that the use of logistic regression models was able to identify the fake news with an accuracy of 71.67% [32]. Fairbank et al. concluded that the use of logistic regression models outperformed better when compared to the random forest in the detection of fake news [33]. Agrawal et al. utilized the support vector machine (SVM) scheme and raking scheme to detect misinformation and concluded that the SVM model outperformed standalone system classification [34]. Prasetijo et al. used the SVM along with stochastic gradient descent (SGD) to detect hoax and concluded that the accuracy of SVM was found to be increased to 4–20% with SGD [35]. Granik et al. utilized the Naïve Bayesian classification and concluded that it was reported 74% accuracy in the detection of fake news [36].

5 Future Perspectives and Conclusions

In the current chapter, we reviewed the overall AI technology and its applications to combat fake news. Among them, mostly used algorithms to control the fake news were machine learning and deep learning models in the current generation. This is because they are integrated with a varied number of different sets of algorithms that can work efficiently and able to predict false information correctly. We conclude that to combat fake news efficiently awareness among the public is also necessary apart from the technology integration in our routine lives.

References

1. Wang CC (2020) Fake news and related concepts: definitions and recent research development. Contemp Manag Res 16:145–174
2. Wisker ZL, McKie RN (2021) The effect of fake news on anger and negative word-of-mouth: moderating roles of religiosity and conservatism. J Mark Anal 9:144–153
3. Brasoveanu A, Moodie M, Agrawal R (2020) Textual evidence for the perfunctoriness of independent medical reviews. In: CEUR workshop proceedings of CEUR-WS, pp 1–9
4. Celliers M, Hattingh M (2020) A systematic review on fake news themes reported in literature. Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics), pp 223–34
5. Trivedi MVM (2014) Role of social networking tool in dissemination of Information at Smt. Hansa Mehta library. Soc Sci Res 2:1–14
6. Zhu H, Wu H, Cao J, Fu G, Li H (2018) Information dissemination model for social media with constant updates. Phys A Stat Mech Its Appl 502:469–482
7. Driscoll LN (1994) A validity assessment of written statements from suspects in criminal investigations using the scan technique. Police Stud Int'l Rev Police Dev. 17:77–88
8. Bogaard G, Meijer EH, Vrij A, Merckelbach H (2016) Scientific content analysis (SCAN) cannot distinguish between truthful and fabricated accounts of a negative event. Front Psychol 7:243
9. Fuller CM, Biros DP, Wilson RL (2009) Decision support for determining veracity via linguistic-based cues. Decis Support Syst 46:695–703
10. Zhou L, Burgoon JK, Nunamaker JF, Twitchell D (2004) Automating linguistics-based cues for detecting deception in text-based asynchronous computer-mediated communication. Gr Decis Negot 13:81–106
11. Buller DB, Burgoon JK (1996) Interpersonal deception theory. Commun Theor 6:201–242
12. Buller DB, Burgoon JK, Buslig A, Roiger J (1996) Testing interpersonal deception theory: the language of interpersonal deception. Commun Theory 6:268–289
13. Pennebaker JW, Booth RJ, Francis ME (2007) Operator's manual: linguistic inquiry and word count—LIWC2007. Depts Ttu Edu 1–21
14. Ali M, Levine T (2008) The language of truthful and deceptive denials and confessions. Commun Rep 21:82–91
15. Rubin V, Conroy N, Chen Y, Cornwell S (2016) Fake news or truth? Using satirical cues to detect potentially misleading news. In: NAACL-CADD 2016 work computing approaches to deceptive detect 15th annual conference on North Am Chapter Assocication Computing Linguist Hum Lang Technology, pp 1–11
16. Zhao Z, Resnick P, Mei Q (2015) Enquiring minds: early detection of rumors in social media from enquiry posts. In: WWW 2015—Proceedings of the 24th international conference on world wide web
17. Mihalcea R, Strapparava C (2009) The lie detector: explorations in the automatic recognition of deceptive language. In: ACL-IJCNLP 2009—Jt conference on 47th annual meet association computing linguist 4th international Jt conference national language process AFNLP, proceeding of the conference on 2009, pp 309–12
18. Ott M, Cardie C, Hancock JT (2013) Negative deceptive opinion spam. In: NAACL HLT 2013—2013 proceedings of the 2013 conference of the North American chapter of the association for computational linguistics: human language technologies, main conference 2013, pp 497–501
19. Rayson P, Wilson A, Leech G (2000) Grammatical word class variation within the British National Corpus Sampler. New Front Corpus Res Pap from Twenty First Int Conf English Lang Res Comput Corpora, Sydney 2002(36):295–306
20. Ott M, Choi Y, Cardie C, Hancock JT (2011) Finding deceptive opinion spam by any stretch of the imagination. ACL-HLT 2011 Proc 49th Annu Meet Assoc Comput Linguist Hum Lang Technol 309–19
21. Johnson M (1998) PCFG models of linguistic tree representations. Comput Linguist 24:614–632

22. Feng S, Banerjee R, Choi Y (2012) Syntactic stylometry for deception detection. 50th Annu Meet Assoc Comput Linguist ACL 2012—Proc Conf 171–5
23. Khan A, Feng J, Liu S, Asghar MZ (2019) Optimal skipping rates: training agents with fine-grained control using deep reinforcement learning. J Robot 1–10
24. Tacchini E, Ballarin G, Della Vedova ML, Moret S, de Alfaro L (2017) Some like it Hoax: automated fake news detection in social networks. CEUR Workshop Proc. 2017, pp 1–15
25. Pratiwi IYR, Asmara RA, Rahutomo F (2018) Study of hoax news detection using naïve bayes classifier in Indonesian language. Proc 11th Int Conf Inf Commun Technol Syst ICTS 2017:73–8
26. Kim J, Tabibian B, Oh A, Schölkopf B, Gomez-Rodriguez M (2018) Leveraging the crowd to detect and reduce the spread of fake news and misinformation. WSDM 2018—Proc 11th ACM Int Conf Web Search Data Min 324–32
27. Habib A, Akbar S, Asghar MZ, Khattak AM, Ali R, Batool U (2018) Rumor detection in business reviews using supervised machine learning. Proc—2018 5th Int Conf Behav Econ Socio-Cultural Comput BESC 233–7
28. Asghar MZ, Rahman F, Kundi FM, Ahmad S (2019) Development of stock market trend prediction system using multiple regression. Comput Math Organ Theory 25:271–301
29. Ma J, Gao W, Wong KF (2017) Detect rumors in microblog posts using propagation structure via kernel learning. ACL 2017—55th Annu Meet Assoc Comput Linguist Proc Conf (Long Pap 2017) 708–17
30. Hamidian S, Diab M (2019) Rumor detection and classification for twitter data [Internet] 2019. Available from: http://bit.ly/c0J2aI
31. Ahmed H, Traore I, Saad S (2017) Detection of online fake news using n-gram analysis and machine learning techniques. Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics) 127–138
32. Ball L, Elworthy J (2014) Fake or real? The computational detection of online deceptive text. J Mark Anal 2
33. Fairbanks J, Fitch N, Knauf N, Briscoe E (2018) Credibility assessment in the news : do we need to read ? Mis2
34. Agrawal T, Gupta R, Narayanan S (2017) Multimodal detection of fake social media use through a fusion of classification and pairwise ranking systems. 25th Eur Signal Process Conf EUSIPCO 2017:1045–9
35. Prasetijo AB, Isnanto RR, Eridani D, Soetrisno YAA, Arfan M, Sofwan A (2017) Hoax detection system on Indonesian news sites based on text classification using SVM and SGD. Proc 2017 4th Int Conf Inf Technol Comput Electr Eng ICITACEE 2017: 45–9
36. Granik M, Mesyura V (2017) Fake news detection using naive Bayes classifier. 2017 IEEE 1st Ukr Conf Electr Comput Eng UKRCON 2017—Proc 2017:900–3

Design of a Chatbot for Four- to Ten-Year-Old Children Based on Emotional Intelligence

Swati Rajwal

Abstract The development of emotional intelligence in children begins during the early years of a child. Although it is the responsibility of parents to help a child in developing emotional awareness, studies have shown the utility of software systems in aiding this process. In this paper, the author presents the design of an emotionally intelligent chatbot for children. The outcomes of an online survey conducted among the parents reported that 70% of the respondents felt that an emotionally intelligent interactive chatbot can be useful for children to cope with intense subject matters related to low grades, no friends, bullies, and others. The study highlights various features of a chatbot like a user interface, personalization, responsiveness, security, and human intervention. From the findings, the author has suggested five design principles along with the detailed architecture of a chatbot framework. The paper will be useful for future studies that seek to design and develop a highly efficient emotionally intelligent chatbot for children which is trusted by their parents.

Keywords Chatbot · Natural language processing · Child · Emotional intelligence · Child–smartphone interaction

1 Introduction

A chatbot is an artificially intelligent software tool that allows the users to communicate and give verbal or textual commands to perform actions like turning off a light, mathematical computations, order pizza, set reminders, booking a flight, and other generic activities. While commercial chatbots are highly popular, emotionally intelligent chatbots are equally on the rise in various domains [1]. Emotional intelligence in chatbots refers to its ability to identify the emotion associated with the user conversation and then generate an appropriate reply. In 1966, Weizenbaum developed a program called ELIZA [2] which was capable of interacting with user via text.

S. Rajwal (✉)
Division of Computer Engineering, Netaji Subhas University of Technology, New Delhi, India
e-mail: swatir.cs20@nsut.ac.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_57

The conversation with ELIZA was therapeutic in nature [1] and people enjoyed the conversation. This implies that a well-designed chatbot has the caliber to emotionally help, or at least assist, people to some extent.

Children as young as two years old are capable of using smartphones [3] and have access to the Internet. Therefore, it is important to protect the privacy and interests of a child. This can be achieved, to some extent, with the help of a chatbot. An emotionally intelligent chatbot can assist a child like a friendly human companion. An open conversation with a chatbot would allow children to express emotions [4] and even deal with issues like having no friend, or de-motivation due to low grades or online bullies [5, 6]. Children can share their stories as many times as they would want to without any fear. Therefore, utmost care should be given while designing a chatbot for children.

The rest of the paper is organized as follows: Sect. 2 discusses the literature review. Section 3 presents online survey methodology and results. Section 4 provides various features of an emotionally intelligent chatbot for children. Section 5 discusses the chatbot design principles. Finally, Sect. 6 concludes the paper.

2 Literature Review

Many researchers have reported the design and implementation of chatbots for children for various useful applications including but not limited to conversation-based tutoring [7], coping with online threats [6] and online abuse [5], privacy identification [8], behavioral coaching [9], children with autism spectrum disorder [1], increasing positive attitude [10], and expressing emotions [4].

Ruan et al. [7] created a chatbot to teach English to six-year-old children. Another chatbot aimed at helping children recognize and express their emotions [4]. For children with special abilities like autism spectrum disorder (ASD), Li et al. [1] designed a chatbot using bidirectional long-short term memory (Bi-LSTM) and an attention mechanism with word embedding. Some researchers combined chatbot with website to help children report online incidences of abuse [5]. These recent studies indicate the great potential that a chatbot has in helping children, especially in times of distress. Table 1 summarizes the recent research studies considered for the literature review.

3 A Survey Among Parents

This section talks about the online survey conducted among parents. Section 3.1 talks about the survey methodology, and Sect. 3.2 analyses the survey results.

Table 1 Summary of literature review

Reference	Contribution
Li et al. [1]	Built a Chinese chatbot using deep learning for children with ASD
Santos et al. [4]	Chatbot design to help identify the emotion associated with the story that the child tells to the bot
Rita et al. [5]	End-to-end chatbot for children to report cybercrimes with ease
Piccolo et al. [6]	Proposed the design requirements of a chatbot for children to deal with online threats
Ruan et al. [7]	Developed a chatbot for Chinese-speaking children to learn English from reading material
Lin et al. [8]	Developed a chatbot-based privacy identification system for children
Ghandeharioun et al. [10]	Reported the design of an emotionally intelligent chatbot and also indicated that extroverts liked such a chatbot more than introverts
Stephens et al. [9]	Developed a behavioral counseling chatbot for adolescents
Mrsic et al. [12]	Design of a chatbot for University student office

3.1 Survey Methodology

A Google Form with 10 short questions was shared among parents of children from four to ten year age groups. Survey was completely anonymous and the parents were informed about the anonymity. Participants were asked for their country of residence, child's age, whether the child uses any digital gadgets, can the child give voice commands, does the parent get frustrated if the child asks the same question over and over again. Questions specially designed to understand sentiments of parents toward an emotionally intelligent chatbot were also asked—do you feel a child can get demotivated and why, are there some issues a child may not want to talk about with parents. Finally, participants were asked for their personal opinions about the features that an emotionally intelligent chatbot must possess. The survey form was released on 25th September 2021 and is open to responses for one week.

3.2 Survey Results

The online survey received responses from 13 parents residing in various geographical locations like India, the USA, and UAE. A surprisingly large number of parents

(69.2%) think that at times a child can indeed get demotivated. When asked about the plausible reasons for the lack of motivation among children, 53.8% of the survey respondents think that having no friends is one of the biggest reasons followed by low grades and bully at school. Moreover, 69.2% of the respondents are affirmative that an emotionally intelligent and interactive chatbot can be helpful to a child to deal with the aforementioned issues. The usability of chatbot is further supported by the fact that 61.5% of the parents believe there can be some issues a child may not want to talk with the parents first [6]. Apart from a chatbot being interactive and fun, about 46.2% of the parents suggested that the chatbot should 'be realistic and a little bit of scolding is good.' Therefore, parents would trust a chatbot that is not just friendly to the child but also rational and realistic at times.

4 Features of the Chatbot

In light of the literature review and the online survey results, various features of an emotionally intelligent chatbot for children are shown in Fig. 1.

- **User Interface (UI)**: The 23 usability principles proposed by Tuli et al. [13] should be considered while planning the UI of a chatbot for children. Additionally, the UI should also include two means of communication. The *voice-based communication* is a convenient way of interacting with a chatbot wherein the child speaks and the sentence will be internally translated into textual form. On the other hand, *text-based communication* allows interactions with the chatbot by sending text messages. Unlike text, voice-based communications can sometimes become ambiguous for the chatbot to understand resulting in unexpected results. Also note that the responses ought to be delivered both textually and verbally by the chatbot, unless specified otherwise.

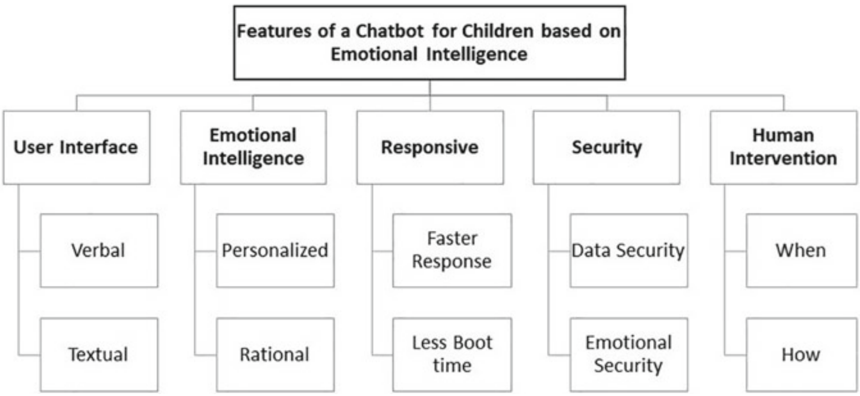


Fig. 1 Various features of a chatbot for children

- **Emotional Intelligence**: It is well-supported by the survey results that *personalization* and *rationality* are the two fundamental aspects of an emotionally intelligent chatbot for children. A chatbot's ability to comprehend child's concern, unstructured at many times, and reply in an exclusive manner is called personalization. This makes the child feel special and valued. Dissimilar to human companions who may sometimes become a bad influence on the child, a chatbot should always talk with a rational mindset. For children's intellectual growth and according to the survey results, 46.2% of the parents suggested that they would trust a chatbot that is not just friendly to their child but also rational and realistic at times.
- **Responsive**: A child may want to talk to the chatbot at any hour of the day. Thus, the chatbot should be available all the time and the *response time* ought to be as quick as could really be expected. Alongside being profoundly responsive, the chatbot should have a very little *boot time*. The Boot time or startup time is the time taken by the chatbot application to open up and be ready for the child to operate.
- **Security**: Chatbot interactions generate a lot of data making it vulnerable to security breaches like cookies, user data, and identification [14], and unconsented use of children's information. With the right system in place, *data security* can be enabled to ensure no user data is compromised at any expense. Apart from data security, *emotional security* should also be taken into the chatbot design considerations. Emotional security guarantees that the chatbot conversations are not threatening in any sense to the child. This would allow the child to express themselves more openly.
- **Human Intervention**: Though researchers have developed chatbots that act like human companions [1, 6, 7] and therapists [4], it is important to acknowledge that a chatbot is an artificially programmed software with inevitable limitations. It is an utmost priority to design a chatbot that is capable of gracefully handling the unknown conversation without creating absurd responses leading to more complications. Figure 2 shows when human intervention is necessary during the conversation between chatbot and child.

5 Design Principles

To develop an end-to-end emotionally intelligent chatbot, the author proposes five design principles as discussed below:

- **Communication**: The chatbot design ought to include verbal and textual means of communication between the child and chatbot. The system/ gadget integrated keyboard and microphone will be used for getting the text and verbal messages, respectively. Unlike humans who may get frustrated with the same questions being asked again and again (69.2% of survey respondents felt they do get frustrated sometimes), the chatbot should be able to respond in a good way and even handle unstructured messages from the child.

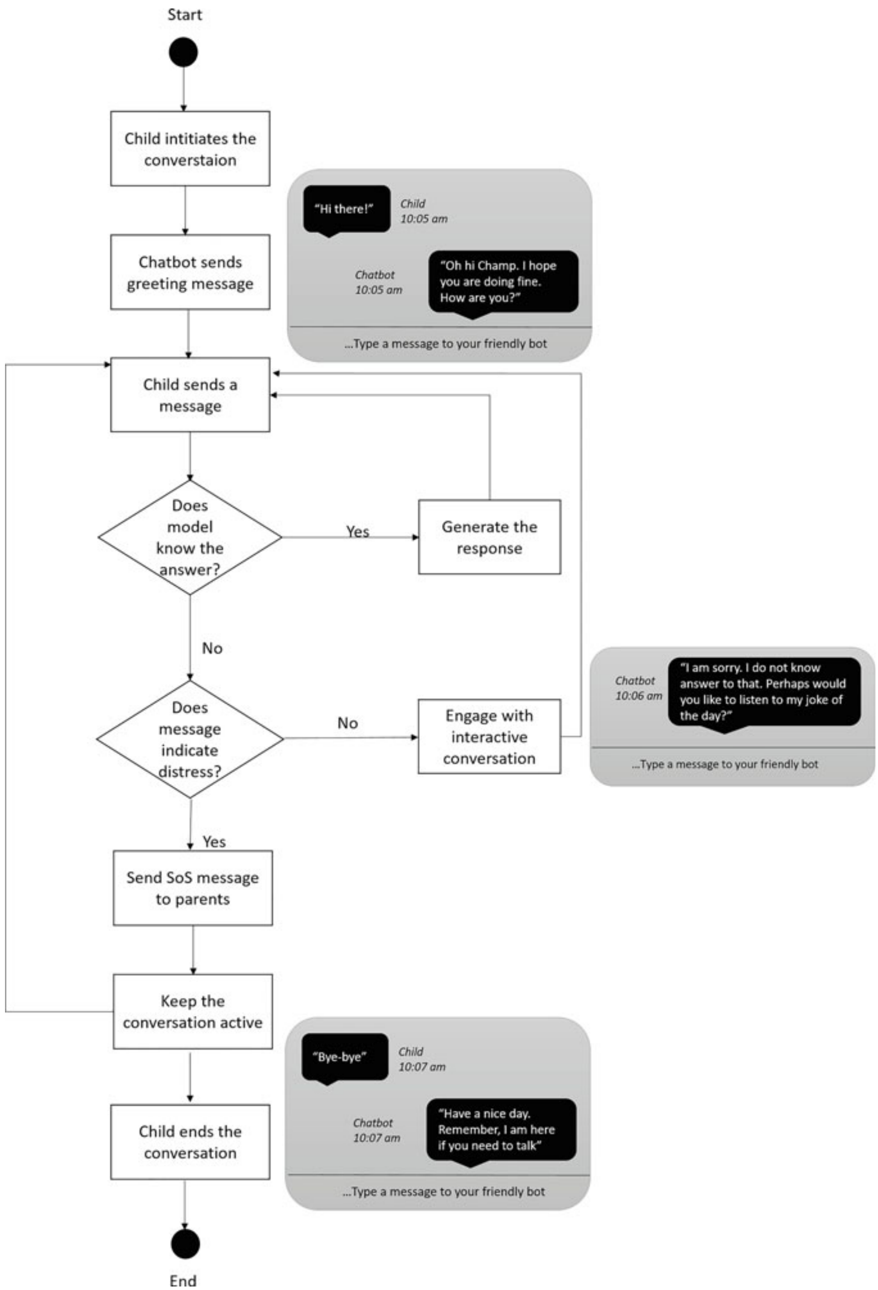


Fig. 2 Flowchart for handling cases where human intervention is required

- **Knowledge Base (KB)**: For the chatbot to understand a child, it needs to be trained upon a dataset or the KB. In the current scenario, a KB consists of various relevant conversations that a child may have with a chatbot. These conversations can be categorized into various topics which helps a chatbot to make better decisions. For instance, a topic of conversation can be 'having no friends at school.' Making a corpus of conversations can be an extensive cycle. But a KB rich in quantity and quality will definitely help the NLP model to learn from the best. Subsequently, the chatbot will be able to respond with the most appropriate answer.
- **NLP Model**: Natural language processing (NLP) deals with analyzing the human language. NLP gives any software the power to 'understand' a human in his/her language, i.e., the natural language. Therefore, NLP is an inevitable part of a chatbot design [10] since at every step the chatbot has to understand the context of the conversation and process the text accordingly to generate an appropriate response [11, 15]. In a chatbot, an NLP model and KB are equivalent to human brain and various past experiences, respectively. Apart from NLP, some researchers have utilized deep learning-based models to train a chatbot [1].
- **Application Programming Interface (API)**: A model trained on the KB can be used to do computations like generating an apt reply to a child's statement. An API provides a neat and convenient abstraction between the actual chatbot application and the trained model. The API consumers only use the functionalities without knowing any implementation. Creating an API will allow the front-end, i.e., chatbot application, to interact with the backend, i.e., trained custom model, to get a response to child's query.
- **Chatbot Application**: This is the front-end or user-facing part of the chatbot. The child will directly interact with this without being aware of how the requests are being served behind the scene. The Chatbot can be a dedicated mobile app, web-based, or even integrated with social media platforms such as LINE [8], Twitter, Instagram, and others. Also, a chatbot developer can use already existing integration systems like Google Dialogflow or design a new one. The functional requirements for the chatbot should include the ones discussed earlier in Sect. 4.

Figure 3 shows interactions among various design components that make a chatbot system. Integration of voice- and text-based conversation features is made easier by various toolkits like Google's DialogFlow and Assistant. Dialogflow provides a text-based UI, and the same can be integrated with Google Assistant to provide voice-based conversation. In both the cases, backend, or the brain of the chatbot, is a trained custom NLP model based on a KB. This brain is responsible for understanding the user statements and generating an appropriate reply. The child initiates the conversation by saying or typing something. The chatbot receives the input which is referred to as a *request* from the child. An API call will be made at the backend along with the request. Upon receiving the API call with the request, the model, i.e., brain, processes the request and generates an appropriate *response*. Thus, all the design principles combined together make the conversation between the child and the chatbot feasible.

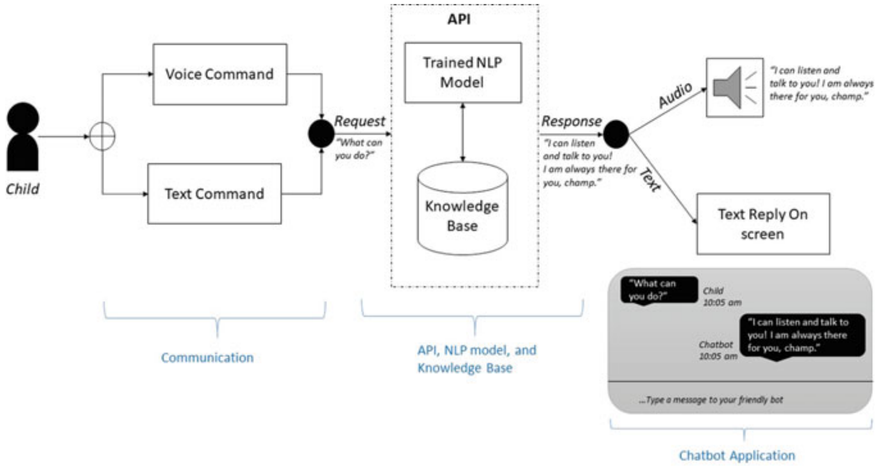


Fig. 3 Chatbot design architecture

6 Conclusion

The present study proposed the design of an emotionally intelligent chatbot for four- to ten-year-old children. The study was conducted based on online survey where experiences of parents from various geographies were gathered. The content analysis of responses from an online survey along with the literature review gave the author an insight into the parent's expectations from an emotionally intelligent chatbot and its usability for children. Various features of an emotionally intelligent chatbot for children were reported in this study. In addition to that, the design principles and their collaboration for such a chatbot were discussed in detail. This study is useful for a wide range of researchers who want to design and develop an emotionally intelligent chatbot that is useful to children as well as trusted by the parents.

References

1. Li X, Zhong H, Zhang B, Zhang J (2020) A general Chinese chatbot based on deep learning and its' application for children with ASD. Int J Mach Learn Comput 10(4):1–10
2. Weizenbaum J (1966) ELIZA–a computer program for the study of natural language communication between man and machine. Commun ACM 9(1):36–45
3. Yadav S, Chakraborty P (2021) Child-smartphone interaction: relevance and positive and negative implications. Univ Access Inf Soc. https://doi.org/10.1007/s10209-021-00807-1
4. Santos KA, Ong E, Resurreccion R (2020, June) Therapist vibe: children's expressions of their emotions through storytelling with a chatbot. In: Proceedings of the interaction design and children conference, pp 483–494
5. Rita MN, Shava FB (2021, Aug) Chatbot driven web-based platform for online safety and sexual exploitation awareness and reporting in Namibia. In: 2021 international conference on

artificial intelligence, big data, computing and data communication systems (icABCD). IEEE, pp 1–5
6. Piccolo LSG, Troullinou P, Alani H (2021) Chatbots to support children in coping with online threats: socio-technical requirements. Designing interactive systems conference 2021:1504–1517
7. Ruan S, Willis A, Xu Q, Davis GM, Jiang L, Brunskill E, Landay JA (2019, June) Bookbuddy: turning digital materials into interactive foreign language lessons through a voice chatbot. In: Proceedings of the sixth (2019) ACM conference on learning@ scale, pp 1–4
8. Lin PC, Yankson B, Lu Z, Hung PC (2019, July) Children privacy identification system in LINE Chatbot for smart toys. In: 2019 IEEE 12th international conference on cloud computing (CLOUD). IEEE, pp 86–90
9. Stephens TN, Joerin A, Rauws M, Werk LN (2019) Feasibility of pediatric obesity and prediabetes treatment support through tess, the AI behavioral coaching chatbot. Transl Behav Med 9(3):440–447
10. Ghandeharioun A, McDuff D, Czerwinski M, Rowan K (2019, Sept) Towards understanding emotional intelligence for behavior change chatbots. In: 2019 8th international conference on affective computing and intelligent interaction (ACII). IEEE, pp 8–14
11. Rahman AM, Al Mamun A, Islam A (2017, Dec) Programming challenges of chatbot: current and future prospective. In: 2017 IEEE region 10 humanitarian technology conference (R10-HTC). IEEE, pp 75–78
12. Mrsic L, Mesic T, Balkovic M (2020) Feb) Cognitive services applied as student support service chatbot for educational institution. International conference on innovative computing and communications. Springer, Singapore, pp 417–424
13. Tuli N, Mantri A (2020) Usability principles for augmented reality based kindergarten applications. Proc Comput Sci 172:679–687
14. Hasal M, Nowaková J, Ahmed Saghair K, Abdulla H, Snášel V, Ogiela L (2021) Chatbots: security, privacy, data protection, and social aspects. Practi Exper Concurr Comput:e6426
15. Ayanouz S, Abdelhakim BA, Benhmed M (2020, March) A smart chatbot architecture based NLP and machine learning for health care assistance. In: Proceedings of the 3rd international conference on networking, information systems & security, pp 1–6

Context-Based Vulnerability Risk Scoring and Prioritization

Dhruv Prashant Shah, Shreyans Munesh Patel, Jainam Vinay Tailor, Shubh Rajiv Kumar Bhagat, and Archana Nanade

Abstract Protecting an organization's intellectual property, financial secrets, and performance is crucial because it is sensitive data that if compromised could be catastrophic to the organization in question. As a result of the growing economy, organizations of scale have a significant portion of their infrastructure over technology which makes the organization vulnerable. The security teams of such organizations work to patch such vulnerabilities as they come across them but may spend a significant amount of organization resources fixing vulnerabilities that may not be exploited. After conducting our own research on the existing methods to prioritize vulnerabilities that have a higher probability of being exploited, we found that machine learning can be used to make the process of vulnerability prioritization efficient. This paper discusses our research on using machine learning for vulnerability prioritization and the different machine learning algorithms that can be of use for the same. This paper also discusses our approach on creating a system for vulnerability prioritization in an organization.

Keywords Vulnerability · CVE · CVSS · ML · SVM · Random forest

1 Introduction

In terms of computing, security includes both cybersecurity and physical security, which businesses utilize to prevent illegal access to data centers and other computerized systems. Information security is a subset of cybersecurity that aims to protect data's confidentiality, integrity, and availability. Cybersecurity can benefit with risk management by preventing cyber-attacks, data breaches, and identity theft.

D. Prashant Shah (✉) · S. Munesh Patel · J. Vinay Tailor · S. Rajiv Kumar Bhagat · A. Nanade
Computer Engineering MPSTME, NMIMS, Mumbai, Maharashtra, India
e-mail: Dhruv076787@gmail.com

A. Nanade
e-mail: archana.nanade@nmims.edu

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_58

Machine learning is a branch of artificial intelligence (AI), where the main objective is to give the computer the ability to learn from a provided set of data. The structure of the data is understood, after which the data is fit into models. These models can be successfully utilized by people for any given application where machine learning is required.

This research will be focusing on harnessing the potential of machine learning alongside the principles and understanding of cybersecurity, in which a model will be trained to identify and prioritize the vulnerability having the highest number of chances of getting exploited.

The contributions of our research are as follows:

- Choosing the most efficient machine learning algorithms over others for purposes of training provided datasets pertaining to cybersecurity.
- A review of various research papers wherein machine learning and vulnerability management tools are used in conjunction with each other.
- The elaboration of our proposed system working and how the same would be beneficial to society.

2 Objectives

With vulnerabilities comes the danger of losing, damaging, or destroying a subject's data. Risk is defined as the possibility of losing, damaging, or destroying assets or data because of a cyber threat. Most organizations today prioritize vulnerabilities using one of two methods: they use the Common Vulnerability Scoring System (CVSS) to choose which vulnerabilities to fix first, or they accept the vulnerability priority offered by their vulnerability assessment tools. Context-based risk scoring is the key to prioritizing a vulnerability. Context-based risk scoring, the feature that surrounds the risk scoring for a particular vulnerability established is called context-based risk scoring / in simple terms the circumstances that form the setting for an event, statement in terms of which it can be fully understood, along with the risk scoring what are the other features present with it.

Features such as the description, category, impact of the vulnerability, target, port, URL location. Organizations ignore the features and mainly prioritize the vulnerability based on risk score, as large the number of vulnerabilities listed is the reason for it. With the help of machine learning, we can achieve accurate results, keeping the risk score and features in consideration.

3 Literature Review

Machine learning algorithms have not only proven to be an essential tool in the field of mathematics and engineering but also in information security as well. In this part of the paper, we discuss the various research papers written by researchers and scholars all around the globe.

In this paper [1], the authors make use of the existing CVE vulnerability descriptions and CVSS ratings from the OSVDB (Open-Source Vulnerability Database), which were used to implement and evaluate the proposed models. Text mining tools were used to extract feature vectors after which Random Forest algorithm, SVM, and fuzzy systems were examined hence forecasting their respective CVSS scores.

In this study [2], the authors provide a score for the severity of vulnerabilities, and CVSS aids in prioritization. CVSS scores lack information on the potential exploit victim's context in their most common application. Researchers and executives in the industry have long recognized that the degree of vulnerabilities varies substantially depending on the environment of the firm. The authors tested their method by putting it to the test on a sample set, consisting of 720 vulnerability notifications from the NVD and making observations that confirmed integrating context information significantly improved the vulnerability response prioritizing and selection process.

The authors of the paper [3] describe a machine learning-based technique for classifying each vulnerability description by kind automatically. We compared our proposed scheme's performance to that of existing algorithms, examined misclassification scenarios, and uncovered the potential for a range of human errors. The authors compared the suggested scheme's performance to that of other algorithms, looked at situations of misclassification, discovered the possibility for various human errors, and attempted to fix them.

In paper [4], the authors illustrate the utility of basic machine learning approaches in forecasting computer operating system security using data from public repositories. The Random Forest technique was found to surpass other algorithms in forecasting computer operating system vulnerability severity levels based on F-measure, recall, and precision.

In the paper [5], the authors leverage past vulnerability data to predict the likelihood of an exploit and the time frame in which the unknown vulnerabilities may occur, in this thesis using machine learning algorithms. According to this study, the most important criteria include common information from external references, vendor products, and vulnerability descriptions. It was found that 83 percent prediction accuracy for binary classification using multiple distinct machine learning techniques was possible. According to the authors, the relative performance of several of the algorithms is negligible in terms of criteria like accuracy, precision, and recall. The authors suggest a linear time support vector machine (SVM) algorithm as the best classifier for both performance metrics and execution time. The exploit time frame forecast demonstrates that relying solely on public or published dates to classify vulnerabilities or exploits is insufficient and concludes that data quality must be improved to obtain better forecasts.

Because of the scarcity of security specialists, labeling reports with vulnerability identifiers had been done manually, which has resulted in human-induced errors and scalability concerns. In this research [6], the authors present a machine learning-based technique for automatically classifying each vulnerability description by kind. We compared the performance of our suggested scheme to that of existing algorithms, looked at situations of misclassification, and discovered the potential for a variety of human errors. The authors experimentally demonstrated the performance of the proposed scheme in comparison with other algorithms, analyzed cases of misclassification, revealed the potential for numerous human errors, and tried to correct them.

This paper [7] introduces the available CVE vulnerability descriptions and their accompanying CVSS ratings from the OSVDB database which were used for the implementation and evaluation of the proposed models. To forecast the concerned CVSS scores, feature vectors were extracted using text mining tools and methodologies, and then, the SVM and Random Forest algorithms, as well as fuzzy systems, were evaluated. Even though SVM and Random Forest are the most widely used and trusted methods for prediction, the findings of this study show that fuzzy systems can produce equivalent or even better outcomes. Furthermore, the fuzzy-based approach is significantly easier and faster to construct.

4 Proposed System

The data we are fetching is from a vulnerability management tool which is the tool which is currently used by and is a proprietary tool of the company we are collaborating with for the data that they will provide. There are certain data points that they would be sending us which we would be weighing for importance before the same is implemented into the algorithm. We are not allowed to disclose the above-mentioned parameters; however, we are describing the process by which the data would be used to derive information that allows us to prioritize a vulnerability over another using the CVSS score as a base (Fig. 1).

These are the steps which we are going to follow.

1. ***Fetch the data***

   This stage deals with the collection of input parameters for the model. A specific set of parameters are collected from a vulnerability management tool that will be fed to our machine learning model.
2. ***Process and evaluate***

   In this stage, we manually process the data. Dropdown all the null values as they are of no use to us and just reduce the processing time of the machine.
3. ***Send processed data to model***

   In this stage, once the data is processed it is sent to our model with all the necessary inputs.
4. ***Run model on the dataset***

   In this stage, we run the dataset with the shortlisted algorithm which are Naïve Bayes, Random Forest, SVM, and decision tree and whichever algorithm gives

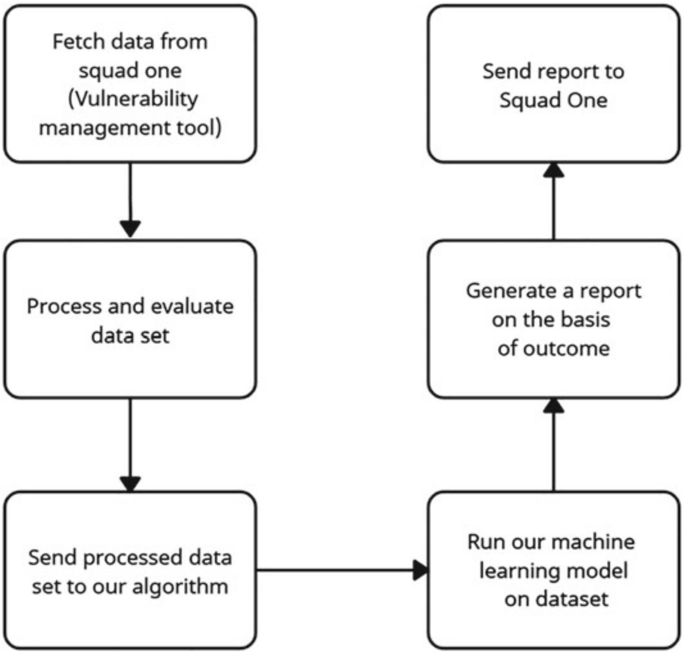


Fig. 1 Flow diagram of the proposed system

the best accuracy on training and the testing test would be finalized for all the future predictions.

5. ***Report generation***

   Once the model gives the result will be generated, which will be sent to the dashboard of the vulnerability management tool so that the analyst can fix the vulnerability with the highest risk first without wasting any time.

5 Proposed Algorithm

5.1 Architecture Diagram

These are the following steps which we are going to follow to get our desired outputs (Fig. 2).

1. ***Input from nessus and qualys***

   Nessus and Qualys are vulnerability management tools that generate the vulnerability report, that includes different features, such as the id, description of the vulnerability, CVSS score, and CVE, and the report from these tools acts as an input for our project.

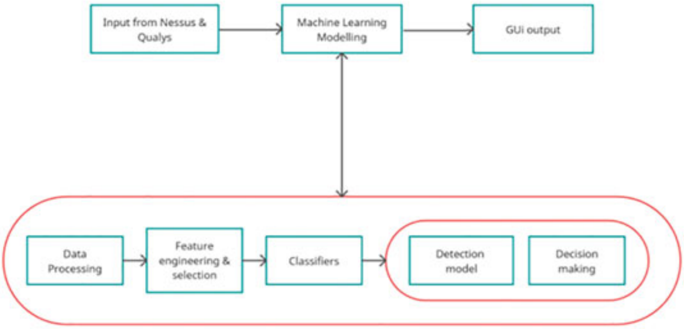


Fig. 2 Architecture diagram of the proposed system

2. ***Data processing***

   The act of performing operations on data particularly by a computer, to extract, transform, or classify data.
3. ***Feature engineering and selection***

   Feature engineering is the process of extracting features from raw data using domain knowledge. A feature is a quality shared by independent units that can be used for analysis or prediction.
4. ***Detection model***

   Detection is a challenging computer vision task that involves predicting both where the objects are in the image and what type of objects were detected.
5. ***Decision-making***

   Detection is a difficult computer vision problem that entails predicting both where and what type of items are detected in a picture.
6. ***Report generation***

   On the output of the given instance, the project creates a report based on the inputs received.

5.2 Random Forest

Random Forest is a supervised learning classification technique and is often considered as an ensemble machine learning method, as it is used for classification, regression, and probability. Random Forest can find missing values from many datasets, and it can provide a more accurate value by creating a forest of decision trees during the learning phase, where the number of trees indicates the robustness of the forest as well as the accuracy of the algorithm. Random sampling is used to train the characteristics for sampling nodes, by combining several decisions to form a single decision.

The algorithm consolidates multiple forests on different subsets of a dataset and averages the results to enhance the performance of the dataset's detection accuracy. The accuracy of the machine can be further improved by combining several classifiers.

Step 1: Select 'p' features at random from a total of features, q.

Step 2: Calculate node 'd' from the selected 'p' features using the best split point.

Step 3: Use the best split and then subdivide the node into daughter nodes.

Step 4: Repeat until the number '1' is reached.

Step 5: Repeating the preceding processes 'n' number of times, until a forest of 'n' trees is built.

Random Forest has been selected as it comparatively takes less time to train.

5.3 Decision Tree

A classification model is like a tree. It builds a structure of nodes and branches from the evidence collected during the learning phase of the model for each attribute. The connection of the nodes and branches is determined by the number of entities in the dataset. Each attribute has a set number of values used by the forwarding process. The decision for each transaction is reached by following the rules described on each node and branch. At last, the class label will be assigned to the records according to the decision node. This procedure has multiple iterations and will repeat till a class category is assigned to each transaction. In all, attributes are converted to nodes and branches and one of them is selected at the decision.

Step 1: Import libraries

Step 2: Import dataset

Step 3: Split the dataset into the training set and test set

Step 4: Train the decision tree regression model on the training set

Step 5: Predict the results

Step 6: Compare the real values with the predicted values

Step 7: Visualizing the decision tree regression results.

5.4 Random Forest Regression

Aggregate of multiple decisions is taken in Random Forest Regression. A complex problem is solved by combining multiple classifiers, which also helps improve the machine's accuracy. Amidst the learning phase, a forest is generated which consists of multiple trees. This classifier consolidates multiple forests for different subsets of a dataset and averages out the results to increase the accuracy of the machine's detection capabilities.

Step 1: Import libraries

Step 2: Import dataset

Step 3: Split dataset into a training set and test set

Step 4: Train Random Forest Regression model on the training set
Step 5: Predict results
Step 6: Compare real values with predicted values
Step 7: Visualize Random Forest Results.

Random forest regression was selected from the referenced papers [6], as it comparatively takes less time to train while providing high accuracy, which increases the efficiency of the model. Random Forest Regression achieved an accuracy of 81% [5].

5.5 Support Vector Machine (SVM)

The objective of the SVM algorithm is to find the optimal line or decision boundary for categorizing n-dimensional space to categorize new data points in the correct category. The best decision boundary is referred to as a hyperplane.

Step 1: Import Python libraries
Step 2: Display image of each bee type
Step 3: Image manipulation with rgb2gray
Step 4: Histogram of oriented gradient
Step 5: Create image features and flatten them into a single row
Step 6: Loop over images to pre-process
Step 7: Scale feature matrix + PCA
Step 8: Split into train and test sets
Step 9: Train model
Step 10: Score model
Step 11: ROC curve + AUC.

The algorithms that have been discussed above when implemented in the same way will in theory give us the results that we are aiming to get; however, the actual implementation of the same will confirm our approach. We plan on implementing our above discussed approach and creating an implementation paper to document the same.

6 Social Implication

It is evident that the dependence of corporate work, media applications and miscellaneous use on technology is spread across a network, which can make a user vulnerable to simply existing on such platforms. This makes it imperative that vulnerabilities be patched on a priority basis. Making sure a vulnerability is worth patching will help entities host user data to save valuable time and resources on the vulnerabilities that matter rather than patching vulnerabilities that may not be exploited. In 2019, Kenna Securities published that out of 18,000 vulnerabilities published only 473 were widely exploited which totals around 2% of the total. So, it is important to patch only those vulnerabilities that are most likely to be exploited. This will allow for a more efficient approach to patches when deployed, thus preventing the system or entity in question from being compromised.

7 Conclusion and Future Work

This paper identifies the need for a new type of vulnerability analysis, i.e., predict severity level of software vulnerability based on CVSS and vulnerability description. This new type of vulnerability analysis can simplify vulnerability management and prioritization for non-security experts because it requires only the 'surface-level' information that describes how a vulnerability works. And in the future, we will implement the entire proposed system with a real database and would try to get the same accuracy.

References

1. Farooq HM, Otaibi NM (2018) Optimal machine learning algorithms for cyber threat detection. In: 2018 UKSim AMSS 20th international conference on modelling and simulation. IEEE
2. Alperin KB, Wollaber AB, Gomez SR (2020) Improving interpretability for cyber vulnerability assessment using focus and context visualizations. In: 2020 IEEE symposium on visualization for cyber security (VizSec), IEEE, pp 30–39
3. Alenezi F, Tsokos CP (2020) Machine learning approach to predict computer operating systems vulnerabilities. In: 2020 3rd International conference on computer applications and information security (ICCAIS), IEEE, pp 1–6
4. Spring JM, Galyardt A, Householder AD, VanHoudnos N (2020) On managing vulnerabilities in AI/ML systems. New Secur Paradigms Workshop 2020:111–126
5. Edkrantz MICHEL (2015) Predicting exploit likelihood for cyber vulnerabilities with machine learning. Master's thesis
6. Aota, Masaki, Hideaki Kanehara, Masaki Kubo, Noboru Murata, Bo Sun, and Takeshi Takahashi. "Automation of Vulnerability Classification from its Description using Machine Learning." In 2020 IEEE Symposium on Computers and Communications (ISCC), pp. 1–7. IEEE, 2020.
7. Khazaei A, Ghasemzadeh M, Derhami V (2016) An automatic method for CVSS score prediction using vulnerabilities description. J Intell Fuzzy Syst 30(1):89–96

Performance Comparison of Machine Learning and Deep Learning Algorithms in Detecting Online Hate Speech

F. H. A. Shibly, Uzzal Sharma, and H. M. M. Naleer

Abstract The main objective of this research is to analyze and compare the performance of machine learning (ML) and deep learning (DL) algorithms in detecting online hate speech. Therefore, Support Vector Machine (SVM), Random Forest (RF), Decision Tree (DT), Logistic Regression (LR), Convolution Neural Network (CNN), Recurrent Neural Network_Long Short-Term Memory (RNN_LSTM), BERT (Bidirectional Encoder Representations from Transformers), and Distil BERT algorithms have been explored and analyzed in this research. This research has applied the dataset on hate speech which was developed by Andry Samoshyn which is publicly available in Kaggle. ML algorithms and DL algorithms have got good scores in accuracy. In ML, SVM, RF, and LR have got top accuracy values. In DL algorithms, RNN_LSTM, Distil BERT, and BERT have performed well in accuracy. Based on F-measurement, DL classifiers have outperformed ML algorithms. Distil BERT has obtained the highest F-measurement scores. When we compare the overall performances, DL is performed well rather than ML in detecting hate speech. Especially transformer-based models of DL are more efficient than other DL and ML algorithms.

Keywords Hate speech · Machine learning · Deep learning Twitter · And performance comparison

F. H. A. Shibly (✉)
Assam Don Bosco University, South Eastern University of Sri Lanka, Oluvil, Sri Lanka
e-mail: shiblyfh@seu.ac.lk

U. Sharma
Department of Computer Applications, School of Technology, Assam Don Bosco University, Guwahati, India
e-mail: uzzal.sharma@dbuniversity.ac.in

H. M. M. Naleer
Department of Computer Science, Faculty of Applied Sciences, South Eastern University of Sri Lanka, Oluvil, Sri Lanka
e-mail: drnaleer@seu.ac.lk

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_59

1 Introduction

As digital communication becomes more integrated into our daily lives and the actual and virtual worlds grow more globalized and varied, new problems such as researching online hate speech emerge on the scientific and policy agendas. With the proliferation of networked digital platforms, the collapsing of public, semi-public, and private settings, and the abundance of data generated by public online discussions, digital social science methods are rapidly embracing computational approaches [1].

People have become more involved with the widespread of social networks over the last few decades. People over a large area now can reveal and spread their ideas, concepts, and opinions immediately via microblogging applications. These types of expressions allow researchers to investigate how people are feeling about a variety of events. So that people can speak freely and share their views in many angles. Internet is one in which it is a reason for the action of harmful and aggressive content, that is not always safe. Prejudice and aggression were often expressed in online hate speech. Disparaging someone based on their sexual orientation, race gender, religion, ethnicity, or nationality is considered hate speech [2].

There are still a lot of unanswered issues about hate speech. For one thing, such kind of harmful speech is punished or if open speech rights of free speech should be extended to it is a hotly contested topic [3–5]. Another area of disagreement is whether repression or measures that target the cause of the issue, such as counter-speech and education, should be employed as a countermeasure [6]. These concerns, on the other hand, are irrelevant in the absence of the ability to identify and categorize hate speech in large quantities.

Speech that is intended to incite hatred is often described as any communication that minimizes someone or a group of people because of their gender identity, race, skin color, sexual preference, ethnic origin, national origin, or religious conviction [7]. Additionally, hate speech may be described as an objectionable way of talking or writing that employs a perspective regarding a particular target people or target group to convey anti-social philosophy [8]. According to both experts' definitions, such kind of speech is any kind of exchanging or spreading ideas that is insulting, denigrates, or humiliates others.

Studies on hate speech on social media have been conducted before and have become fascinating to debate. Related research used English text data to classify and identify hate speech on Twitter [9]. Although many studies have recently been conducted on the detection and control of such hate speech, no definitive solution has yet been reached. To this end, several methods such as ML, DL, and pre-trained models of artificial intelligence are being studied. Continuing research efforts are needed, however, as to what mechanisms effectively identify such hate speech. In that sense, our study aims to find a suitable mechanism by measuring and comparing the performance of machine learning, ML, and transformational models over to Twitter dataset. This study covers selected ML, DL, and transformational model algorithms from the literature review. The literary review reveals that such comparative studies have been carried out to a lesser extent, that only ML and DL have been compared, and that transformational models have not yet been included in such comparisons. This comparative study of ours will play an important role in identifying and controlling hate speech on social media itself, as previous studies have evaluated the effectiveness of each individually. Therefore, our paper makes the following significant contributions:

- Exploring and identifying the most common and suitable algorithms for detecting hate speech in ML and DL models for this research.
- Calculating the performance of selected algorithms over to the Twitter dataset based on F1-score, precision, recall, accuracy, confusion matrix, and ROC curve matrices.
- Comparing the performance of each algorithm.
- Finding and suggesting the most efficient algorithm(s) in detecting online hate speech.

It has become a necessity nowadays to detect and control hate speech at the management and social levels. There are various problems in the society due to such hate speech. It is having a huge impact on everything from personal relationships to family relationships. Such problems in social systems severely affect many and hinder multifaceted social progress. That is why the main task of the community is to manage the hate speech that can be found in the posts placed on social media and remove the inappropriate speech.

2 Literature Review

Classification algorithms can do the detection after the text has been prepared for machine use. Classifiers used in ML can be divided into three types: supervised, semi-supervised, and unsupervised. A vast volume of text must be manually labeled. Hence, this supervised learning is domain specific. A supervised approach is most often used to detect hateful language in texts. It is shown that using supervised classifiers to detect hate speech on Twitter, Burnap, and Williams [10] found that all classifiers performed the same as each other in terms of accuracy.

According to a study [11], both Facebook and Twitter are used to target people based on their characteristics. This is due to a lack of consistent enforcement and regulation. Using the Weka library, this research examines the impact of different subsets of characteristics on classification. It conducted stemming tests to see how stemming affected the categorization task. Support Vector Machines (SVM), J48, which is a supervised technique for building a Decision Tree (DT), Naive Bayes (NB), and another decision tree algorithm Random Forest were all used to solve the classification issue, according to the researchers (RF). Finally, this study discovered that in this setting, the NB and linear SVM models performed much better than the other models.

Natural language analysis and ML techniques like Naive Bayes (NB) approach were used to examine hate speech aimed against African Americans in [12]. Reference [13] studied how well ML classifiers classified cyberhate utilizing n-gram-based methods against certain kinds of hate speech. The source [14] classified generic hate speech using character and word n-grams as characteristics and BLR as the classification technique. A few notable deep learning neural network techniques for identifying hate speech in tweets have also been developed that outperform existing models [15]. There were three distinct methods utilized in this study, including Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and FastText, all of which were combined with random or GloVe word embeddings to get the desired result: CNN, LSTM, and FastText. The authors of Zhang and Luo (2018) [16] were able to distinguish between social media content that was racist, sexist, or not hostile depending on whether the material was racist, sexist, or not hostile.

Current researchers have successfully used the CNN model to a variety of text-related problems, including natural language processing and some more techniques. References [17–19], spam filtering, and others. Text mining tasks will benefit appreciably from ML models in the future. However, it has additional depth and is more dependent on artificial neural networks. It attempts to mimic events using layers of neurons and learns to look for patterns in the input text. ML procedures are not necessarily preferable to more standard supervised methods. As with any other ML technique, ML's performance depends on the algorithm and the number of hidden layers used. Al-Smadi et al. [23] validated the above premise by comparing the performance of RNN and SVM. As a result, they recommended using a different approach for embedding in future studies. Instead of using a word embedding to implement features, Pitsilis et al. [24] employed an RNN model with word frequency vectorization to remove language reliance in terms of embedding approaches for hate speech identification. For hate speech identification, their approach outperformed existing state-of-the-art ML methods.

Because of recent advancements in machine learning, better sentence representations have been developed. The use of RNNs allowed for the modeling of lengthier text sequences. The complete use and applications of the algorithm of RNNs, such as LSTMs [25] and GRUs [26], allowed for the improvement of the representation of long-term dependencies. Because of this, LSTM and CNN-based models outperformed character- and word-based n-gram models in terms of classification performance, by a significant margin. [27] Character-based modeling utilizing Char-CNNs has been used to hate speech classification [28], and it has been shown to be effective.

Moreover, there is another model that has started to get attention in this field. There are some pre-trained libraries available to test many datasets. Such pre-trained model is called transformational models, and there are few research works available in the field of detecting hate content. We may create robust and semantically rich embeddings using approaches such as Bidirectional Encoder Representations from Transformers (BERT) [29], which have been trained on large amounts of data, and then utilize them for downstream tasks such as hate speech detection. The widespread use of BERT has spurred the development of a few algorithms that are built on the

BERT [30] architecture. There are numerous of these classifiers, and some of the most famous include Roberta, Distil BERT [31], and XLNET [32], among others. BERT is an upgrade on this algorithm that is practiced and instructed using a larger dataset to improve results, and when compared to the more complex BERT method, Distil BERT utilizes a smaller dataset and is optimized for better results.

Pretraining using generalized autoregressive techniques aims to recover the original data from corrupted input [33]. It may be described as follows: For this reason, the literature has been reviewed to determine the processes of ML and transformational models, and suitable algorithms have been chosen for this research. As a result, the SVM, RF, DT, and LR classifiers have been identified from the ML classifiers. This study has also investigated and used CNN, RNN LSTM from DL, BERT, and Distil BERT from DL transformational models.

3 Methodology

Under the methodology section, researchers explain the overall method and dataset that were used in this research, as well as the collection and structuring of the experimental dataset and setup, among other things.

3.1 Experimental Dataset

For this study, we used the Hate Speech and Offensive Language dataset, which was created by Andry Samoshyn and made freely accessible on Kaggle [34]. A dataset based on Twitter data is being used to investigate hate speech identification. It includes 24,783 English text messages from Twitter, each of which has been classified into one of the three categories listed as neutral, offensive, or hateful.

3.2 Experimental Setup

This study used Python to accomplish the methods that were proposed. The Keras software program was used to apply the LSTM architecture. A PC running Windows 10 with an Intel(R) Core (TM) i5-8265U processor operating at 1.60 and 1.8 GHz, 8 GB of RAM, and a 1 TB hard disk drive was used to conduct the tests.

3.3 Pre-processing

The casual nature of Twitter discussions results in data that is highly unstructured and includes a large amount of noise, which may reduce the accuracy of the method used to analyze it. Because of this, it was decided that all Tweets should be pre-processed to delete less detective features. It is commonly identified that pre-processing increases the efficiency of classification algorithms [35] while simultaneously reducing the processing time required to perform the classification. Lowercasing and stemming were used to minimize word inflexions, and stopwords were removed from the labeled dataset at the time of normalization.

3.4 Proposed Method

The suggested technique for classifying Twitter messages into three classes is detailed in this part: "hate speech," "offensive language," and "neither hate speech nor offensive language." In this research, the methodology for categorizing tweets into three categories is explained in this section. As shown in Fig. 1, the whole study process was followed. On this diagram, the five key stages in the research method are shown. The procedures utilized in this study were data collection, pre-processing, training the models with chosen algorithms, testing the models separately, and evaluating each model, which are all depicted in the figure.

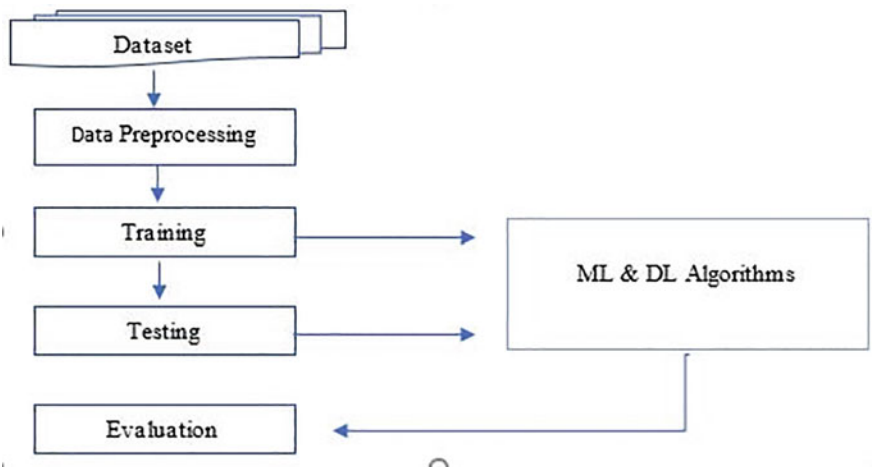


Fig. 1 Proposed method

3.5 Evaluation

In the disciplines of data mining and data retrieval, evaluating the accuracy of selected classifiers is one of the most important phases. Error rate and F-measure are widely used to determine the accuracy of a classifier's ability to locate the proper category or class of unknown cases. The error rate is the instances of the test set that were erroneously categorized. We will call this set of data "X" and let "m" represent how many occurrences were misclassified by a classification model C. You can calculate the accuracy of C in selecting the correct classes of X instances using the following formula:

$$\text{Accuracy(C)} = \frac{m}{n} \tag{1}$$

The error rate approach ignores the cost of inaccurate predictions in ML. For the most part, F-measure is used to solve this problem. To determine the value of F-measure, two basic metrics are used: precision and recall. Imagine that some of the data in the test set belong to a certain class or category S. It assigns a category label to each test data. There will be four kinds of forecasts for the test set S.

Percentage of accurately forecast data for category S is known as precision. Percentage of correctly forecast real data for category S is known as recall. It is possible to calculate the F-measure based on precision and recall (2–4).

$$\text{Precision} = \frac{|TP|}{|TP| + |FP|} \tag{2}$$

$$\text{Recall} = \frac{|TP|}{|TP| + |FN|} \tag{3}$$

$$\text{F - measure} = 2\cdot \neq \frac{\text{Precision.Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

4 Results and Discussions

4.1 Parameter Evaluation

In this analysis, all algorithms were tested by using precision, recall, F-measures, and accuracy. Based on the dataset, hate speech (HS), offensive languages (OL), and no hate speech (N) classifications were tested. But this research focuses on hate speech only.

As soon as the top classifiers had been identified, the following step was to evaluate the performance of each model on a set of test data. When comparing the performance of different ML algorithms, a wide variety of performance metrics are utilized. When dealing with classification problems, the confusion matrix and area under curve (AUC) are often used. DL systems are also evaluated based on their accuracy, recall, specificity, and F1-score, among other metrics. ROC curves are helpful for displaying the performance of classifiers since they are easy to understand. The connection between the sensitivity and specificity of a trained classifier is represented by the receiver operating characteristic curve (ROC curve).

4.2 Estimation of F-Measures

Regardless of class imbalance, the F-measures are regarded as one of the finest metrics for classification models. The F1-score is a weighted average of the class's recall and accuracy. Its best and worst values are 1 and 0. Additionally, it is a measure that combines accuracy and recall, often known as the conventional F-measure or balanced F-score. In this research, Distil BERT has got highest F1-score, and it is outperformed other algorithms (Table 1).

4.3 Comparison of F-Measures and Accuracy Between ML and DL

According to Fig. 2, ML algorithms have got good accuracy. But DL algorithms have also got good scores in accuracy. In ML, SVM, RF, and LR have got top accuracy values. In DL algorithms, RNN_LSTM, Distil BERT, and BERT have performed well in accuracy. Based on F-measurement, DL algorithms have outperformed ML algorithms. Distil BERT have obtained highest F-measurement scores. When we compare the overall performances, DL is performed well rather than ML in detecting hate speech. Specially, transformer-based models of DLs are more efficient than other DL and ML algorithms.

There were few research works available similar to this context, and the findings can be compared with such papers. Paul and Bora [37] had conducted a research on deep learning algorithms in detecting multilingual hate. They have used LSTM and BiLSTM algorithms and found that the scores calculated for accuracy, precision, and F1-score suggest that LSTM has performed better than BiLSTM. Aluru et al. [38] found that the algorithm of mBERT outperformed other selected algorithms. Another research [39] suggested that CNN model gave the best performance, with an F1-score. But, in this research, we proposed that BERT and LSTM algorithms can perform well than CNN models to detect hate contents. Al-Hassan and Al-Dossari [40] have approached with 4 deep learning models: LTSM, CNN + LTSM, GRU, and CNN + GRU and concluded that adding a layer of CNN to LTSM enhances the overall performance of detection. From these comparisons, we can come to a

Table 1 Measurements of algorithms

Algorithms	Parameter evaluation				
	Precision	Recall	F-Measure	Accuracy	
SVM	HS	0.68	0.16	0.26	
	OL	0.93	0.97	0.95	91%
	N	0.96	0.91	0.88	
RF	HS	0.65	0.15	0.24	
	OL	0.92	0.97	0.94	91%
	N	0.84	0.89	0.87	
DT	HS	0.35	0.29	0.32	
	OL	0.93	0.94	0.93	89%
	N	0.85	0.85	0.85	
LR	HS	0.62	0.16	0.26	
	OL	0.91	0.97	0.94	90%
	N	0.86	0.83	0.85	
CNN	HS	0.34	0.15	0.21	
	OL	0.86	0.95	0.90	83%
	N	0.73	0.53	0.61	
RNN_LSTM	HS	0.41	0.30	0.35	
	OL	0.92	0.95	0.94	89%
	N	0.87	0.84	0.86	
BERT	HS	0.37	0.38	0.37	
	OL	0.91	0.94	0.93	88%
	N	0.90	0.75	0.82	
DISTIL BERT	HS	0.38	0.42	0.40	
	OL	0.93	0.93	0.93	88%
	N	0.90	0.75	0.82	

decision that DL algorithms are better than ML in detecting hate speech from various perspectives.

5 Conclusions and Future Research

In this research paper, researchers try to use the selected ML and DL algorithms to detect online hate speech. This study identified four ML algorithms including SVM, RF, DT, and LR and four DL algorithms including CNN, RNN_LSTM, BERT, and Distil BERT based on the literature review. All algorithms have been undergone into performance analysis using F1-score, precision, recall, accuracy, confusion matric,

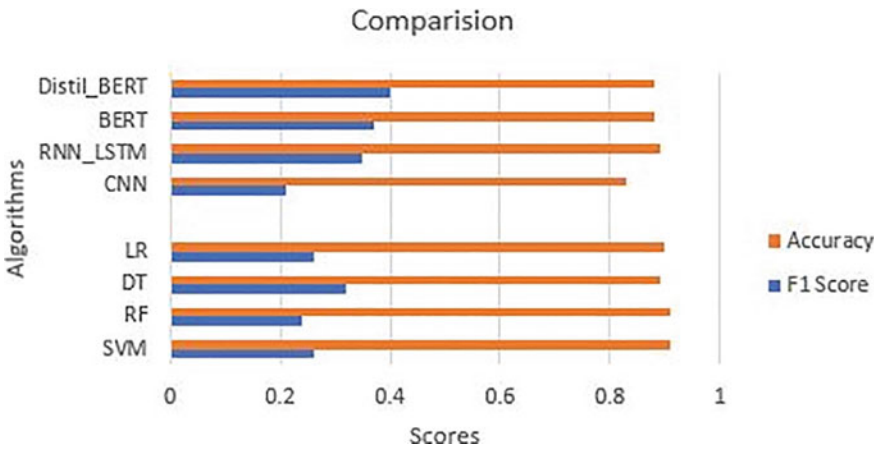


Fig. 2 Comparison of F-measures

and ROC curve metrics to find out their preferences over to the Twitter hate speech dataset. It is revealed that all algorithms have good accuracy in detecting hate speech. Specially, SVM and RNN_LSTM have got highest accuracy in ML and DL, respectively. But, Distil BERT has outperformed all other algorithms in detecting hate speech and it performed well in all metrics. Since it is a pre-trained DL algorithm, its performances are highly commendable and another pre-trained model called BERT is also performed well. Therefore, we suggest that pre-trained or transformer-based DL algorithms will be more efficient in detecting online hate speech. In our future research works, we will incorporate multilingual datasets to compare the ML and DL algorithms to get more efficient results and other transformer-based models are also to be applied further.

References

1. MacAvaney S, Yao HR, Yang E, Russell K, Goharian N, Frieder O (2019) Hate speech detection: challenges and solutions. PLoS ONE 14(8):e0221152
2. Fortuna P, Nunes S (2018) A survey on automatic detection of hate speech in text. ACM Comput Surv (CSUR) 51(4):1–30
3. Barendt E (2019) What is the harm of hate speech? Ethic Theo Moral Prac 22. https://doi.org/10.1007/s10677-019-10002-0
4. Dworkin R (2006) A new map of censorship. Index Censorship 35(1):130–133. https://doi.org/10.1080/03064220500532412
5. Heyman S (2009) Hate speech, public discourse, and the first amendment. In: Hare I, Weinstein J (eds) Extreme speech and democracy. Oxford Scholarship Online. https://doi.org/10.1093/acprof:oso/9780199548781.003.0010
6. Brown A (2018) What is so special about online (as compared to online) hate speech? Ethnicities 18(3):297–326. https://doi.org/10.1177/1468796817709846

7. Nockleby JT (2000) Hate speech. Encyclopedia of the American Constitution (pp 1277–1279). Macmillan
8. Warner W, Hirschberg J (2012) Detecting hate speech on the world wide web. In: LSM '12 Proceedings of the second workshop on language in social media (pp 19–26). Montreal, Canada: Association for Computational Linguistics
9. Putri TTA, Sriadhi S, Sari RD, Rahmadani R, Hutahaean HD (2020) A comparison of classification algorithms for hate speech detection. In: IOP conference series: materials science and engineering (Vol 830, No 3, p 032006). IOP Publishing
10. Burnap P, Williams ML (2015) Cyber hate speech on Twitter: an application of machine classification and statistical modeling for policy and decision making. Policy Internet 7(2):223–242
11. Şahi H, Kılıç Y, Sağlam RB (2018) Automated detection of hate speech towards woman on Twitter. In: 2018 3rd international conference on computer science and engineering (UBMK), 2018, pp 533–536. doi: https://doi.org/10.1109/UBMK.2018.8566304
12. Dinakar K, Jones B, Havasi C, Common sense reasoning for detection prevention and mitigation of cyberbullying. dl.acm.org
13. Burnap P et al (2015) Detecting tension in online communities with computational Twitter analysis. Technol Forecast Soc Change 95:96–108
14. Burnap P, Williams ML (2016) Us and them: identifying cyber hate on Twitter across multiple protected characteristics. EPJ Data Sci 5(1):11
15. Woda M, Torbiarczyk M (2018) Use of distributed machine learning toolkit for searching content promoting hate speech on the Web. In: International conference on dependability and complex systems (pp 536–544). Springer
16. Watanabe H, Bouazizi M, Ohtsuki T (2018) Hate speech on twitter: a pragmatic approach to collect hateful and offensive expressions and perform hate speech detection. IEEE Access 6:13825–13835
17. Kim Y (2014) Convolutional neural networks for sentence classification. In: 2014, arXiv:1408.5882. [Online]. Available: http://arxiv.org/abs/ 1408.5882
18. Sadr H, Pedram MM, Teshnehlab M (2019) A robust sentiment analysis method based on sequential combination of convolutional and recursive neural networks. Neural Process Lett 50(3):2745–2761
19. Roy PK (2020) Multilayer convolutional neural network to filter low quality content from Quora. Neural Process Lett 51(1):805–821
20. Akhter MP, Jiangbin Z, Naqvi IR, Abdelmajeed M, Mehmood A, Sadiq MT Document-level text classi_cation using single-layer multisize _lters convolutional neural network," IEEE Access, vol. 8, pp. 42689_42707, 2020.
21. Dordevic JP (2020) The sociocognitive dimension of hate speech in readers' comments on serbian news Websites. Discour Context Media 33:100366
22. Zheng J, Zheng L (2019) A hybrid bidirectional recurrent convolutional neural network attention-based model for text classification. IEEE Access 7:106673–106685
23. Al-Smadi M, Qawasmeh O, Al-Ayyoub M, Jararweh Y, Gupta B (2018) Deep recurrent neural network versus support vector machine for aspect-based sentiment analysis of Arabic hotels' reviews. J Comput Sci 27:386–393
24. Pitsilis GK, Ramampiaro H, Langseth H (2018) Effective hate-speech detection in Twitter data using recurrent neural networks. Appl Intell 48(12):4730–4742
25. Sutskever OV, Le QV (2014) Sequence to sequence learning with neural networks. In: Advances in neural information processing systems, 2014, pp 3104–3112
26. Chung J, Gulcehre C, Cho K, Bengio Y (2014) Empirical evaluation of gated recurrent neural networks on sequence modelling. arXiv preprint arXiv:1412.3555
27. Badjatiya P, Gupta S, Gupta M, Varma V (2017) Deep learning for hate speech detection in tweets. In: Proceedings of the 26th international conference on world wide web companion, WWW '17 companion, international world wide web conferences steering committee, Republic and Canton of Geneva, CHE, 2017, p 759–760. URL: https://doi.org/10.1145/3041021.3054223

28. Zhang X, Zhao J, LeCun Y (2015) Character-level convolutional networks for text classification. In: Advances in neural information processing systems, 2015, pp 649–657
29. Devlin J, Chang MW, Lee K, Toutanova K (2018) Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805
30. Liu Y et al (2019) Roberta: a robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692
31. Sanh V, Debut L, Chaumond J, Wolf T (2019) DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv preprint arXiv:1910.01108
32. Yang Z, Dai Z, Yang Y, Carbonell J, Salakhutdinov RR, Le QV (2019) Xlnet: Generalized autoregressive pretraining for language understanding. In: Advances in neural information processing systems, 2019, pp 5753–5763
33. Mutanga R, Naicker N, Olugbara OO (2020) Hate speech detection in twitter using transformer methods. Int J Adv Comput Sci Appl 11:01
34. Samoshyn A (2020) Hate speech and offensive language dataset (version 01) [Dataset using twitter data, is was used to research hate-speech detection. The text is classified as: hate-speech, offensive language, and neither]. https://www.kaggle.com/mrmorj/hate-speech-and-offensive-language-dataset/code
35. Uysal K, Gunal S (2014) The impact of preprocessing on text classification. Inf Process Manage 50(1):104–112
36. Ting KM (2010) Confusion matrix. Encyclopedia of machine learning. Boston: Springer
37. Paul C, Bora P (2021) Detecting hate speech using deep learning techniques. Int J Adv Comput Sci Appl 12(2). https://doi.org/10.14569/ijacsa.2021.0120278
38. Aluru S, Mathew B, Saha P, Mukherjee A (2021) Deep learning models for multilingual hate speech detection. [online] arXiv.org. Available at: https://arxiv.org/abs/2004.06465v2, Accessed 14 Dec 2021
39. Alshalan R, Al-Khalifa H (2020) A Deep learning approach for automatic hate speech detection in the Saudi Twittersphere. Appl Sci 10(23):8614. https://doi.org/10.3390/app10238614
40. Al-Hassan A, Al-Dossari H (2021) Detection of hate speech in Arabic tweets using deep learning. Published, Multimedia systems. https://doi.org/10.1007/s00530-020-00742-w

A Survey on Various Approaches to Examine Cognitive Behavior and Academic Performance of Learner in Virtual Learning

Rakshit Khajuria, Ashok Sharma, Anuj Sharma, and Parveen Singh

Abstract A virtual learning environment (VLE) is the type of environment that can attract more students because it allows them to study anywhere in the world, which means that the student's location is no longer a constraint. In addition, VLE facilitates access to teaching resources, which facilitate the monitoring of teacher activities and interaction between students and teachers. Therefore, the online environment can assess the factors that lead to an increase or decrease in the academic performance of students. Machine Learning approaches are used for the cognitive behavior and academic performance of students in Virtual Learning. There is still no decision on the parameters to be adopted for the evaluation of virtual teaching as each student may submit the same type of assignment and same Practical files, and can have the same attendance. In such a case, evaluation of a student's academic performance became tough. So we need to adopt some LMS which records various actions of the learners and the teachers like Quiz Submitted On-time/Late, Number of Assignment Submitted On-time/Late, Number of Discussions attended, Number of CA attended, and Practical Submitted On-time/Late, Internet connectivity, etc. So, there is a need for a framework that accounts for all of these parameters' consideration so that a Predictive model can be designed for Forecasting/estimation performance of students that are recommended system should be framed for enhancing the academic performance of the learner.

Keywords Academic performance · Cognitive behavior · Deep learning · Federated learning · Machine learning · Virtual learning

R. Khajuria · A. Sharma
Lovely Professional University, Phagwara, Punjab, India
e-mail: anuj.20592@lpu.co.in

A. Sharma (✉)
University of Jammu, Jammu, India
e-mail: drashoksharma@hotmail.co.in

P. Singh
Cluster University, Jammu, India

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_60

1 Introduction

1.1 Cognitive Engagement

It is the process in which knowledge is absorbed by the users. It includes concentration, stimulating curiosity, concentration, and fluency. Cognitive absorption refers to a state of deep participation, while flow refers to concentrating on one activity without paying attention to any other. CBT is a psychosocial intervention designed to improve mental health [1]. CBT focuses on changing cognitive distortions, behaviors, and challenges, improves emotional regulation, and develops personal coping strategies aimed at solving current problems.

Cognitive behavioral theory is a treatment method that teaches us how to recognize unhelpful or negative thought and behavior patterns. Many specialists believe it is a form of psychotherapy. It intends to assist us in recognizing and exploring the ways in which our emotions and thoughts influence your actions. We can start learning to reframe your ideas in a more positive way as well as helpful ways once you detect this pattern. We all know that due to the impact of COVID-19 our whole economy gets disturbed and it also has a great impact on our education system and due to which has an adverse effect on students [2]. So in the time of the pandemic, the contribution of information technology became the backbone for the education system. With the help of IT solutions, it provides a great platform to connect students and educators face to face on an online mode and the main purpose of online learning is to minimize the learning gap affected during a pandemic situation in COVID-19 [3].

1.2 Behavior of Students in Online Classroom

Online classes are proving to be quite advantageous to many students. Since the students are being handed over the materials to study at the comfort of their home, the productivity level is most likely to rise. Studying at home also means that the students save up on the time which they used to travel to their school or coaching institutes. Students can use this time to adjust their extra studying sessions. Another advantage of online classes is that the students are allowed to have flexibility in their studies [4]. Some students are also not able to study well in pressurizing situations, so studying at home provides them with a comfortable environment in which they can progress well. We can experience these common feelings and thoughts.

1.3 Impact of COVID-19 on Students and Teachers

The education system is highly affected by COVID-19 due to which variations can be seen in it resulting in jolting teachers in great numbers. This write-up expresses the thoughts of teachers bounced back via three need gap analysis, i.e., access, use, and the breach in teaching skills [5]. Data shows that inconsistency linking private as well as public institutions is increasing in weight with the movement to e-learning. The reason is students who are economically challenged and are from far-flung areas are difficult to reach and employees do not know how to make it up for those students as well as for those who have been brutally hit hard by the pandemic [6]. The data also expresses the limitation of not receiving proper online pedagogy training. Ed-Tech arrived, declaring itself to be a fully fledged remedy to all diseases with even more devastating upshots for one's career, classroom work, and subsistence. Nevertheless, Ed-Tech is inappropriate considering difficult homeschoolers who distribute difficult communities. This write-up firstly displays the vocal pedagogy of employees hit hard by the epidemic and then severely inspects whether Ed-Tech plays a crucial role in filling the online educational void [7].

1.4 LMS (Learning Management Systems)

Learning management systems too have seen rapid growth in user traffic post-pandemic. Given the immense importance of such systems and their indispensability, it is pertinent to understand how these tools are helping learners around the world [8].

LMS is an application used to manage, deliver, track, and report learning programs. It provides the framework that manages all aspects of the learning process—it is where you store, deliver, and track your training content. LMS is a tool that can be used to create, manage, and in addition deliver online courses as well as programs. It gives a platform for students and instructors to learn and demonstrate their abilities whenever and wherever they choose. LMS covers most of the major markets like schools, educational institutions, businesses, health care, etc. It helps identify communication gaps between instructors and learners by checking each individual's progress on tests and assessments. [7]. Such virtual e-learning systems have become all the more important in today's digital era. Even before the onset of the pandemic virtual learning platforms made quality learning accessible to people bereft of it. LMS has changed the learning outlook of individuals all over the world. It allows users to choose and learn in the field they want to become good at, as all information can now be shared around the world. There are various types of LMS solutions available today with different development and hosting arrangements: Software as a Service: hosted in the cloud, so no equipment to maintain or upgrade to manage. Self-hosted solution requires you to download and install software on your own devices, but offers more control over dashboard customization and branding. Desktop/mobile apps: Accessible on multiple devices for easier collaboration. Mobile apps are also available [9]. Table 1 provides a summarized view of the LMS, Open Source/Paid, features, and learning analytics.

The objective of this paper Survey on Various Approaches to examine Cognitive Behavior and Academic Performance of Learner in Virtual Learning is we have on a

Table 1 Various LMS used in online learning

S. No.	LMS	Open source/paid	Features	Learning analytics	References
1	Moodle	Open Source	LMS is fully design and stretchable to perfectly meet our needs and integrate with the solution like Microsoft Office 365, Google Apps, etc.	NO	[10]
2	Edmodo LMS	Open source	Its ease of use and accessibility are very suitable for K12 programs	NO	[10]
3	Google classroom	Open source	The platform can help teachers create courses, submit assignments, communicate with students, grade coursework, and post comments, all in one place	No	[23]
4	Canvas	Open source	LMS is part of a digital learning solution, which includes a powerful course builder, dashboard, testing engine, etc.	YES	[24]
5	Schoology	Not open source	Its strength lies in its focus on building and connecting learning communities from students to managers	NO	[23]
6	LearnDash	Open source	LearnDash is a powerful plug-in that allows you to create, manage, modify, and publish courses directly to the popular content management system	No	[24]

survey on various virtual learning platforms and Cognitive Behavior and Academic Performance of students in Virtual learning.

- It presents different LMS used in online learning.
- It provides information about Cognitive Behavior and Academic Performance of Learner in Virtual Learning.

The paper organized as follow: Sect. 2 present Literature Review. Section 3 present Outcomes of survey conducted. Section 4 presents Machine Learning and Virtual learning. Section 5 presents Future Scope. And the last section concludes this paper.

1.5 Federated Learning

Federated Learning allows mobile phones to develop a shared prediction model cooperatively while retaining all of the training data on the device, effectively divorcing machine learning from the requirement to store data in the cloud. Federated Learning enables smarter models, lower latency, and lower energy use while maintaining privacy. This approach also has another immediate benefit: in addition to giving an update to the shared model, the enhanced model on your phone may be used right away, allowing us to create experiences tailored to our preferences [25] (Fig. 1).

FL differs from distributed ML in that the information sent to the server by each participant is a trained sub-model rather than the original data. Asynchronous transmission is also possible with the FL [26].

Classification of Federated Learning

Data is dispersed among the participants in the form of isolated islands in FL, and each participant can represent its own data with a matrix. FL is now divided into three categories based on the distribution of data feature space and sample ID space: horizontal federated learning, vertical federated learning, and federated transfer learning [27] (Figs. 2 and 3).

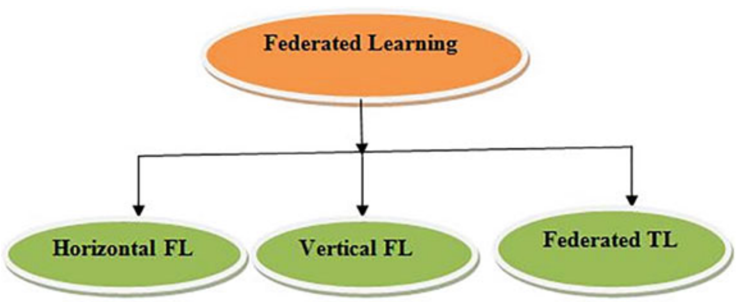


Fig. 1 Categories of federated learning

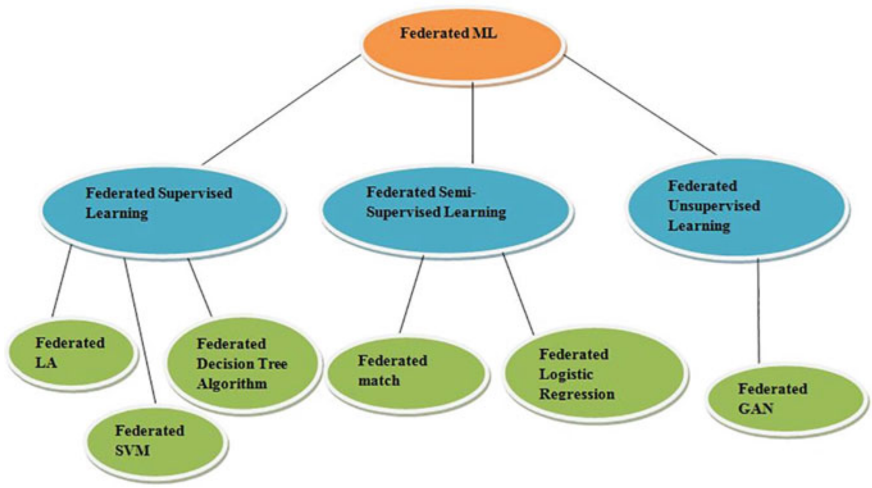


Fig. 2 Classification of federated machine learning

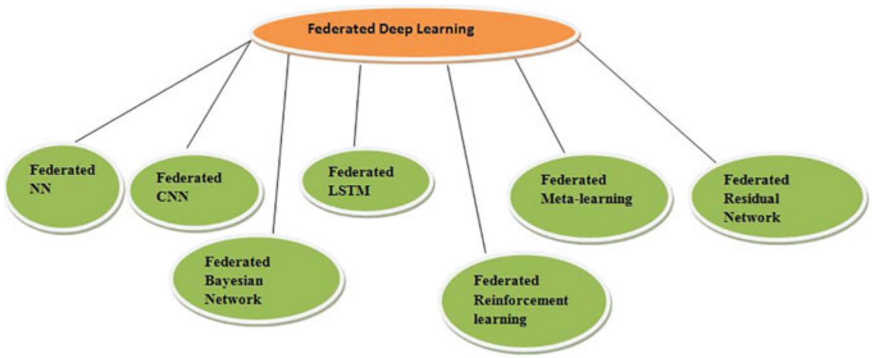


Fig. 3 Classification of federated deep learning

2 Literature Review

The focus of this essay is on students' cognitive behavior and academic performance in virtual learning using Deep Learning and Machine Learning. We have gone through several research papers and the outcome belongs with these results as being presented in this section.

Engr. Bhutto et al. [1] have worked on Comparison of Sequential Minimal Optimization Algorithm vs. Logistic Regression for predicting student's Academic Performance. They have worked on Kalboard 360 which contains 500 records and 16 distinct attributes. The accuracy achieved is more in sequential minimal optimization, i.e., 79% than the logistic regression, i.e., 73%. Martínez et al. [2] have worked on K-means and Unsupervised Clustering approach to Measure the Influence of Behavioral and Personality Factors on Academic Performance of Higher Education Students. They have worked on a Dataset of 153 college freshman students. The accuracy of this model is 80%. Alshehhi et al. [3] have worked on Artificial Intelligence, Online Learning, and Learning Organization for analyses of the impact of AI in online learning during COVID-19. They have worked on Develop a framework for data collection and analysis during COVID-19 and Datasets from Academic References. The output shows that Artificial Intelligence gives good results. Kokoc and Altun [4] have worked on a Prescriptive learning approach combined with an e-learning environment vs. Artificial Neural Network Algorithms for analysis effect of learners interact with learning dashboards on academic performance. They have worked on dataset consists of 126 students enrolled for the 12-week course. The result of the proposed model shows artificial neural network algorithms perform best in predicting academic performance.

Aydoğdu [5] have worked on Deep Learning and ANN for student's performance in online learning toward the end. They have worked on 3518 students of the University. The accuracy of this model is 80.47%. Altuwairq et al. [6] have worked on Convolution Neural Network and Navies Bayes Classifier for Interaction and presence in student learning in an online environment. They have worked on a dataset of 12,271 real-world images from Real-world affective faces (RAF) and 35,887 images of Facial Expression recognition (FER2013). The accuracy of this model is 93%. Conijnet al. [7] has worked on LMS and Predictive models to analyze student performance from learning management system data. They have worked on 17 blended courses with 4989 students. Finally, the result shows the regression analysis of the LMS data. They are available every week to determine whether early intervention is a reasonable possibility. Mubarak et al. [8] have worked on Input–Output Hidden Markov Model (IOHMM), Logistic Regression, and Machine Learning for students prediction early dropouts based on their interaction logs in the online learning environment. They have worked on the dataset taken from (OULA). The accuracy of this model is 84%.

Xu et al. [9] have worked on Machine Learning, Neural Network, and Support Vector Machine for Academic performance prediction related to Internet usage behavior. They have worked on the Internet usage data of 4000 students. The result of the proposed model shows that behavior discipline plays important role in academic success. Redmond et al. [10] have worked on five parameters: Social Commitment, Cognitive Commitment, Behavioral Commitment, Collaboration Commitment, and Emotional Commitment for Online Participation in Higher Education. They have worked on datasets taken from online models. The result along with process was shared with national as well as international experts in online teaching and learning experts, and comments were solicited and received. Ali Akber Dewan et al. [11] have worked on Machine Learning, SVM (Gabor), MLR (CERT), and Boost (BF) for engagement detection in online learning. They have worked on dataset including 112 individuals where 80 males and 32 females. The accuracy of the propose model is MLR (CERT) = 0.714, Boost (BF) = 0.728, and SVM (Gabor) = 0.729. And classifier achieved the correlation of 0.275, 0.329, and 0.306. Ali Mubarak et al. [12] have worked on Deep learning, Support Vector Machine, and Predictive Model (LSTM) for Predictive learning analytics in MOOCs' courses videos. They have worked on a dataset which is taken online from (CAROL). The accuracy of this model is 90%.

Waheed et al. [13] have worked on deep artificial neural networks, SVM, and Logistics Regression for the Academic performance of students based on VLE big data. They worked using a dataset collected from OULA that included 32,593 students over the span of nine months in 2014–2015. The accuracy of this model is 88.5%. Liu et al. [14] have worked on the CNN model, Deep Leaning, and SVM model for Online Classroom Atmosphere Assessment System for Evaluating Teaching Quality? They have worked on Taking sample data online. This model shows a good output. Oreški et al. [15] have worked on Machine Learning, LMS, and Data Mining for Machine learning approaches on LMS data. They have worked on data which is taken from the University of Zagberg. The result of the proposed model shows NN modeling can better classify students than other ML approaches. GE et al. [16] have worked on Machine Learning and EDM for predicting students' academic performance. They have worked on data samples were collected from the Computer Science Department University of Nigeria which consists of 103 first-year students. The accuracy of this model is 92%.

Jiang et al. [17] have worked on Deep Neural Network and Betty brain using Feature Engineering for identity which is better sensor-free affect detection. They have worked on the dataset of 6th class students of an urban public school. The result shows the accuracy of both the proposed models is the same. Zhang et al. [17] have worked on Edge detection, LGCP feature extraction, AWLGCP and FSR method, CLBR-SRC, Gabor SVM, and Active Shape model-SVM for learning trends of University Students during the COVID-19 pandemic. They have worked on a Dataset containing 49,920 labeled images of 47 individuals. The accuracy of this model is 94%. Li et al. [18] have worked on supervised ML algorithms, Cognitive engagement, and Facial behaviors for The Art of Staying Engaged in Problem-Solving: Automated Detection of Cognitive Engagement and they have worked on Dataset consists of 61 students. The results revealed that engaged and less engaged states were detected in 82 and 85 segments, respectively. Everaert et al. [19] have worked on Deep Learning and Surface Learning for The relationship between motivation, learning approaches, academic performance, and time spent. They have worked on Data from questionnaires from 246 students. The result of the proposed model shows 19% of students score poorly on both deep and surface learning methods.

Botelho et al. [20] have worked on Deep learning and Descriptive statistics for Improving Sensor-Free Affect Detection. They have worked on data taken from 15 Australian Schools which consists of 551 students taught by 37 teachers. The result shows that the sample distribution is normal, except that classroom mastery, teacher enthusiasm, and behavioral engagement are negatively skewed. Jim Wu et al. [21] have worked on Mobile-based CRS technology for student's entrepreneurship learning experience using classroom response system. They have worked on dataset consists of an 18-week course in Entrepreneurship Management which consists of 22 graduate students enrolled in it. The results show that CRS technology based on mobile devices is very useful as well as effective tool to promote interaction among students. Waheed et al. [22] have worked on Machine Learning and Deep ANN for Deep Learning Models for Predicting Student Academic Performance from VLE Big Data. They have worked on a sample dataset taken from Open University Learning Analysis (OULA). The accuracy of this model is 89. Kumar et al. [13] have worked on Machine Learning, MATLAB, Mean Square Error, and Threshold-based segmentation for the analysis of student's performance in virtual learning. They have worked on the dataset taken from http://invertory.data.gov/datase. The result of the proposed model indicates that the performance of the artificial neural network is better than that of the support vector machine.

3 Outcomes of Survey Conducted

Mostly used techniques are Machine Learning, Artificial Neural Network, LMS, and Support vector machines. Mostly used dataset is taken from Online learning platforms like MOOCs and from Open University learning analysis (OULA). Mostly used parameters are Number of Quiz Submitted On-time/Late, Number of Assignment Submitted On-time/Late, Number of Discussions attended, Number of CA attended, and Practical Submitted On-time/Late. Mostly, the accuracy of the proposed methodologies is between 85 and 92%. Table 2 provides a summarized view of the approaches/techniques, dataset, result, and gap identity.

4 Machine Learning and Virtual Learning

The principle of machine learning is employed in educational activities. Machine mechanization being utilized or can be diversified in drilling, for instance, implying variance in analyzing choices to such an extent which results in scholar picking excellent opportunities for them and also keeping in mind, inter-patient variability among scholars. Intelligent retrieval can exploit a mystifying course.

Virtual help is an important part of education and an excellent place to apply machine learning. Students can have a dialogue with a virtual assistant.

In this regard, conversational agents provide assistance to students through an application or a web site. The technique is simple, with the pupil just typing text. The agents, on the contrary, carry out the task and determine the right reaction to the input before presenting an easy-to-understand response to the student. Yet, educators acknowledge, by what means knowledge engineering process clarifies as well as revamp the task's coherence. Furtherance built for the acquisition of expert system in the field of study undergoes cut corners for twain, lecture room ventures as well as non-lecture room ventures. Collaborators honored such unrepeatable high ground that constructs training attainable as well as bewitching.

Table 2 Shows approaches used, dataset of images, results, and gaps identifying

S. No.	Authors name	Year	Approaches/techniques	Dataset	Result	Gaps identify	Ref. no
1	Engr. Sana Bhutto et al.	2020	Sequential minimal optimization algorithm versus logistic regression	Kalboard 360. Contains 500 records and 16 distinct attributes	Accuracy achieved is more in sequential minimal optimization, i.e., 79% than the logistic regression, i.e., 73%	Transformation from conventional to e-learning system is difficult to analyze the learning behavior of students	[1]
2	R. Martínez et al.	2019	K-means unsupervised clustering approach	Dataset of 153 college freshman students	The accuracy of the proposed model is 80%	We have to use more dataset in future work	[2]
3	Dr. Abdulmunem Alshehhi et al.	2021	Artificial intelligence Online learning learning organization	During COVID-19, create a framework for data gathering and analysis Academic references' datasets	AI gives good results	To improve organizational results, OL can assist the learning system within the organization	[3]
4	Mehmet Kokoc and ArifAltun	2021	Prescriptive learning approach combined with an e-learning environment versus artificial neural network algorithms	Dataset consists of 126 students enrolled for the 12-week course	Artificial neural network algorithms perform best in predicting academic performance	Large-scale research may focus on assessing the long-term impact of LD on learning performance in open online courses. Only the frequency of PLD exchange data is recorded	[4]

(continued)

Table 2 (continued)

S. No.	Authors name	Year	Approaches/techniques	Dataset	Result	Gaps identify	Ref. no
5	Şeyhmus Aydoğdu	2020	Deep learning Artificial neural network	Dataset consists of 3518 students of University	The accuracy of the proposed model is 80.47%	The number of layers with varied parameters and the number of neurons (deep learning) detected in the hidden layer were both checked to enhance the accuracy score	[5]
6	Khawlah Altuwairq et al.	2021	Convolution neural Network navies Bayes classifier	Datasets consist of 12,271 real-world images from real-world affective faces (RAF) and 35,887 images of Facial Expression recognition (FER2013)	The accuracy of the proposed model is 93%	We must analyze the physical movements and emotions of the students to define their complete behavior of the students. Integrate this method into an intelligent online learning system	[6]
7	Rianne Conijn et al.	2016	LMS predictive model	17 blended courses with 4989 students	Finally, we show the regression analysis of the LMS data. They are available every week to determine whether early intervention is reasonable Possibility	Requires early feedback on the use of learning propensity and LMS data to draw conclusions	[7]

(continued)

Table 2 (continued)

S. No.	Authors name	Year	Approaches/techniques	Dataset	Result	Gaps identify	Ref. no
8	Ahmed A. Mubarak et al.	2020	Input–output hidden Markov model (IOHMM) Logistic Regression Machine Learning	The dataset is taken from (OULA)	The proposed model achieves an accuracy of 84%	We will build more interpretable student behavior models, provide educators with curriculum feedback on student status, such as "how is it going?", and intervene at-risk students based on student behavior and information of background	[8]
9	Xing Xu et al.	2019	Machine learning neural network Support vector machine	Internet usage data of 4000 students	Behavior discipline plays important role in academic success	We will include more data on online behavior	[9]

(continued)

Table 2 (continued)

S. No.	Authors name	Year	Approaches/techniques	Dataset	Result	Gaps identify	Ref. no
10	Petrea Redmond et al.	2018	Social commitment, cognitive commitment, behavioral commitment, collaboration commitment emotional commitment	Dataset is taken from online models	The results along with process were shared with national as well as international experts in online teaching and learning experts, and comments were solicited and received	We need to create an online learning environment to improve learning and teaching outcomes, provide students with opportunities for online participation, and promote connections between themselves, teachers, educational institutions, and industry as they develop strong subject and multi-disciplinary knowledge and skills	[10]
11	M. Ali Akber Dewan et al.	2019	Machine learning SVM (Gabor) MLR (CERT) Boost (BF)	Dataset taken at DAiSEE Dataset include 112 individuals where 80 male and 32 female	Accuracy of MLR (CERT) = 0.714 Boost (BF) = 0.728 SVM (Gabor) = 0.729 And classifier achieved the correlation 0.275, 0.329, and 0.306	Promote online education technology to participate in examination	[11]
12	Ahmed Ali Mubarak et al.	2021	Deep learning Support vector machine Predictive model (LSTM)	The dataset was taken online from (CAROL)	This model shows an accuracy of 90%	We intend to develop a platform for the teaching policy environment	[12]

(continued)

Table 2 (continued)

S. No.	Authors name	Year	Approaches/techniques	Dataset	Result	Gaps identify	Ref. no
13	Hajra Waheed et al.	2020	Machine Learning Deep ANN	Sample dataset taken from Open University Learning Analysis (OULA)	The accuracy of proposed model is 89%	We intend to investigate the importance of activities and identify activities that have an impact on performance	[13]
14	Can Liu et al.	2018	CNN model Deep leaning SVM model	Taking sample data online	This model shows good output	The proposed system can be associated with an inverted classroom shape to make the system more perfect	[14]
15	DijanaOreški et al	2019	Machine learning LMS Data mining Classification	The dataset is taken from University of Zagberg	NN modeling can better classify students than other ML approaches	We will investigate the significance of our research and explore the results of predictive models in the context of teaching	[15]
16	Okereke GE et al.	2020	Machine learning EDM	Data samples were collected from Computer Science department University of Nigeria which consists of 103 first-year student	The accuracy of the proposed model is 92%	The choice of a classifier does not determine the accuracy of prediction but the nature of dataset	[16]
17	Yang Jiang et al.	2018	Deep neural network Betty brain using Feature Engineering	Dataset taken from 6th class students of an urban public school	Accuracy of the two approaches used is same	Explore the generalizability of our findings	[17]

(continued)

Table 2 (continued)

S. No.	Authors name	Year	Approaches/techniques	Dataset	Result	Gaps identify	Ref. no
18	Zhaoli Zhang et al.	2020	Edge detection LGCP feature extraction AWLGCP and FSR method CLBR-SRC Gabor SVM Active shape model-SVM	Dataset containing 49,920 labeled images of 47 individuals	The accuracy of the proposed model is 94%	We can establish corresponding screening models for students with different learning styles and develop adaptive algorithms to integrate all models to evaluate students' learning participation	[17]
19	Shan Li et al.	2021	Supervised ML algorithms Cognitive engagement Facial behaviors	Dataset consists of 61 students	The results revealed that engaged and less engaged states were detected in 82 and 85 segments, respectively	Homogeneity of participants in terms of race and academic background	[18]
20	Patricia Everaert et al.	2017	Deep learning Surface learning	Data from questionnaires from 246 students	19% of students score poorly on both deep and surface learning methods	Accounting educators should also encourage students to do a lot of work during the school year	[19]
21	Anthony F. Botelho et al.	2017	Deep learning Recurrent neural networks EDM	Taken from the ASSISTments learning platform	The accuracy of the proposed model is 75%	By student geographical factors, the differences is introduced	[20]

(continued)

Table 2 (continued)

S. No.	Authors name	Year	Approaches/techniques	Dataset	Result	Gaps identify	Ref. no
22	Yen-Chun Jim Wu et al.	2019	Mobile-based CRS technology	Dataset consists of an 18-week course in Entrepreneurship Management which consists of 22 graduate students enrolled in it	The results show that CRS technology based on mobile devices is very useful as well as effective tool to promote interaction among students	To improve students' academic performance, they need to become responsible and self-controlling when using mobile technology along with services in the classroom to maintain meeting-related activities	[21]
23	HajraWaheed et al.	2020	Deep artificial neural network SVM Logistic regression	The data was obtained from OULA, which included 32,593 students over a nine-month period in 2014–2015	The results show the accuracy of the proposed model is 88.5%	Using natural language processing and advanced deep learning models to extract text data related to student feedback, investigate the importance of activities, and identify activities that impact performance	[22]
24	Mukesh Kumar et al.	2019	Machine learning MATLAB Mean square error Threshold-based segmentation	Dataset used is taken from http://invertory.data.gov/datase	The result shows that the performance of the artificial neural network in is better than that of the support vector machine	Possibility includes changing the total number of neurons or changing successful parameters	[6]

5 Future Work and Conclusion

Later knowledge engineering approach, characteristically of guidance, would fabricate ground breaking Expert System apparatus. Compounded chat-bots imaginably designed multitudinously, expanding urbanity of viro-statics. With the mixture of AI and knowledge engineering, and that comprise uncountable computing along with system analysis, it would give hike affixing cutting edged structure including potentiality in accordance with alteration and grasping, plus predict methods solely. Amalgamation of updated breakthrough along with submerged vast datum would be resorted for forthcoming projects against these methodologies. In the future, we have to prepare for any other pandemic like COVID-19 because due to COVID-19, there is only online learning, no blended learning, and integrity in the online examination is also challenging. There is still no decision on the parameters to be adopted for the evaluation of virtual teaching as each student may submit the same type of assignment, same Practical files, and can have the same attendance. In such a case, evaluation of a student's academic performance became tough. So we need to adopt some LMS which records various actions of the learners and the teachers like Quiz Submitted On-time/Late, Number of Assignment Submitted On-time/Late, Number of Discussions attended, Number of CA attended, and Practical Submitted On-time/Late, Internet connectivity, etc. So, there is a need for a framework that accounts for all of these parameters consideration so that a Predictive model can be designed for Forecasting/estimation performance of students that are recommended system should be framed for enhancing the academic performance of the learner.

Artificial Intelligence along with machine learning has opened up great opportunities within various domains. It has proven surprisingly true within the confines of tutoring industry along with different occupations related to training. Due to this, upcoming edification domains tend to extremely distinguished, permitting trainees realizing their prospective in an eminent or superior way. Intelligent Retrieval is going to be gradually embraced and variegated in different areas of interest. Their bump on ultimate consumer be unsurely evident instead remarkable with regard to premature facet. Multiple clients work together in FL to solve standard distributed ML issues under the supervision of a central server, without having to share their local private data.

Acknowledgements This paper is outcome of JKDST-sponsored project.

References

1. Bhutto ES, Siddiqui IF, Arain QA, Anwar M (2020) Predicting students' academic performance through supervised machine learning. In: 2020 international conference on information science and communication technology (ICISCT), IEEE, pp 1–6
2. Martínez-Rodríguez RA, Alvarez-Xochihua O, Victoria ODM, Arámburo AJ, Fraga JÁG (2019) Use of machine learning to measure the influence of behavioral and personality factors on academic performance of higher education students. IEEE Lat Am Trans 17(04):633–641
3. Alshehhi A, Mansoor W, Alshehhi MA, AlMulla H, Mansoor MD (2021) Impact of artificial intelligence on online learning during Covid-19: a framework. Psychol Educ J 58(2):9581–9587
4. Kokoç M, Altun A (2021) Effects of learner interaction with learning dashboards on academic performance in an e-learning environment. Behav Inf Technol 40(2):161–175
5. Aydoğdu Ş (2020) Predicting student final performance using artificial neural networks in online learning environments. Educ Inf Technol 25(3):1913–1927
6. Altuwairqi K, Jarraya SK, Allinjawi A, Hammami M (2021) Student behavior analysis to measure engagement levels in online learning environments. Sign Image Video Process 1–9
7. Conijn R, Snijders C, Kleingeld A, Matzat U (2016) Predicting student performance from LMS data: a comparison of 17 blended courses using Moodle LMS. IEEE Trans Learn Technol 10(1):17–29
8. Mubarak AA, Cao H, Zhang W (2020) Prediction of students' early dropout based on their interaction logs in online learning environment. Interact Learn Environ 1–20
9. Xu X, Wang J, Peng H, Wu R (2019) Prediction of academic performance associated with internet usage behaviors using machine learning algorithms. Comput Hum Behav 98:166–173
10. Redmond P, Abawi LA, Brown A, Henderson R, Heffernan A (2018) An online engagement framework for higher education. Online Learn 22(1):183–204
11. Dewan MAA, Murshed M, Lin F (2019) Engagement detection in online learning: a review. Smart Learn Environ 6(1):1–20
12. Mubarak AA, Cao H, Ahmed SA (2021) Predictive learning analytics using deep learning model in MOOCs' courses videos. Educ Inf Technol 26(1):371–392
13. Waheed H, Hassan SU, Aljohani NR, Hardman J, Alelyani S, Nawaz R (2020) Predicting academic performance of students from VLE big data using deep learning models. Comput Hum Behav 104:106189
14. Liu C, Ge J, Chen D, Chen G (2018) An online classroom atmosphere assessment system for evaluating teaching quality. In: 2018 IEEE international conference of safety produce informatization (IICSPI), IEEE, pp 127–131
15. Oreški D, Hajdin G (2019) A comparative study of machine learning approaches on learning management system data. In: 2019 international conference on control, artificial intelligence, robotics and optimization (ICCAIRO). IEEE, pp 136–141
16. Okereke GE, et al (2020) A machine learning based framework for predicting student's academic performance. Phys Sci Biophys J 4(2):000145
17. Jiang Y, Bosch N, Baker RS, Paquette L, Ocumpaugh J, Andres JMAL, Biswas G (2018) Expert feature-engineering vs. deep neural networks: which is better for sensor-free affect detection? In: International conference on artificial intelligence in education. Springer, pp 198–211
18. Zhang Z, Li Z, Liu H, Cao T, Liu S (2020) Data-driven online learning engagement detection via facial expression and mouse behavior recognition technology. J Educ Comput Res 58(1):63–86
19. Li S, Zheng J, Lajoie SP, Wiseman J (2021) Examining the relationship between emotion variability, self-regulated learning, and task performance in an intelligent tutoring system. Educ Technol Res Develop 1–20
20. Everaert P, Opdecam E, Maussen S (2017) The relationship between motivation, learning approaches, academic performance and time spent. Acc Educ 26(1):78–107
21. Botelho AF, Baker RS, Heffernan NT (2017) Improving sensor-free affect detection using deep learning. In: International conference on artificial intelligence in education, Springer, pp 40–51
22. Wu YCJ, Wu T, Li Y (2019) Impact of using classroom response systems on students' entrepreneurship learning experience. Comput Hum Behav 92:634–645

23. Kumar M, Singh AJ, Handa D (2019) Performance analysis of students using machine learning and data mining approach. Int J Eng Adv Technol 8:75–79
24. Sharma A, Kumar G, Kumar R (2020) A review on evolution of massive open online courses and their impact on the learning. J Seybold Rep 15(8):1441–1460
25. Sharma A, Singh P (2017) Learning management system for virtual teaching and learning PY. World Acad J Eng Sci 4(1):05–07
26. Shokri R, Shmatikov V (2015) Privacy-preserving deep learning. In: Proceedings of the 22nd ACM SIGSAC conference on computer and communications security, pp 1310–1321
27. Li T, Sahu AK, Talwalkar A, Smith V (2020) Federated learning: challenges, methods, and future directions. IEEE Signal Process Mag 37(3):50–60
28. Kairouz P, McMahan HB, Avent B (2019) Advances andopen problems in federated learning. https://arxiv.org/abs/1912.04977

Intrusion Detection System Using Machine Learning Approach: A Review

Kapil Sharma, Meenu Chawla, and Namita Tiwari

Abstract The abundance of technology followed by serious cybercrimes makes way for providing better security. Internetworking applications at a paramount level generate the need of securing host system as well as network system from the user having malicious intent. Several enterprises and organizations become victims of these severe attacks in the aspect of using many applications for providing safety like firewall, data encryption, and user authentication. By taking into account, this causes a detection system with the use of machine learning approaches of artificial intelligence which has been developed, known as an intrusion detection system (IDS). An intrusion is a process of entering into the system having the intent of unsolicited duplication, record alteration, and illegal access to confidential resources. Hence, analyzing the network packets of such cases for possible intrusions in near future is intrusion detection. The intrusion detection system has evolved as a vibrant topic for researchers in the last two decades and hope it will be in the future as well because of rapid advancement in technology day by day. In this study, a survey of various research papers has been depicted and it has been the utmost priority that the display of work shall be comprehensible.

1 Introduction

In the proportion of advancement fear of unlawful activities has also been increasing rapidly. An attack on the basic pillars of security confidentiality, integrity, and availability is a sequence of activities having the aim of weakening computer network security. There is no system which is made perfectly safe and secure because of few limitations, and hence, the attacker finally finds a loophole in the system to intrude, to analyze the network data for the probable intrusions (attacks).

K. Sharma (✉) · M. Chawla · N. Tiwari
Maulana Azad National Institute of Technology, Bhopal, India
e-mail: kskapilsharma777@gmail.com

M. Chawla
e-mail: chawlam@manit.ac.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_61

Traditional strategies of intrusion prevention like access control schemes, encryption methods, or firewalls have failed to prove in effectively protecting systems and networks from increasingly sophisticated attacks. IDS has become the practical solution and crucial component of any security infrastructure to identify the threats before they produce any widespread damage [15]. An IDS is a software, hardware, and protocols or their fusion responsible for uncovering the possible intrusions from the network finding data [14]. The intrusion detection system is classified into two parts based on detection method and based on data sources, based on detection method it has been further classified into two parts known as signature-based and anomaly-based.

1.1 A Signature-Based Intrusion Detection System (SIDS)

This is better known as the knowledge-based detection and misuse detection technique [12]. In this method, previous attacks pattern is stored in a database for matching of upcoming packets and generates alert accordingly if signature is matched. In SIDS, host's log is checked to find a pattern of activities which have been identified as malicious previously [18]. SIDS generally gives an outstanding accuracy of detection for previously known intrusions [11]. While for SIDS detecting zero-day attacks are very difficult, SIDS has been deployed in several common tools, for instance, NetSTAT, and Snort [12]. The occurrence of zero-day attacks frequently makes SIDS weaken because no prior pattern exists in the database for such attack [4]. To overcome this problem, the possible solution is the anomaly detection technique.

1.2 Anomaly-Based Intrusion Detection System (AIDS)

This model is created by using machine learning statistical and knowledge-based methods, any difference between the model's behavior and observed behavior is considered as an anomaly. During the model building of this model in the training phase, the model is trained by normal traffic profile, and usually testing is done by using the different data sets to build the system's ability for detecting novel attacks. Since it does not rely on the pattern existing in the database, its ability to detect zero-day attack is better than SIDS [2].

1.3 Intrusion Detection Based on the Data Sources

There are two types of IDS systems based on the input data sources, host-based IDS (HIDS) and network-based IDS (NIDS). Detailed classification of data source has been depicted in Fig. 1.

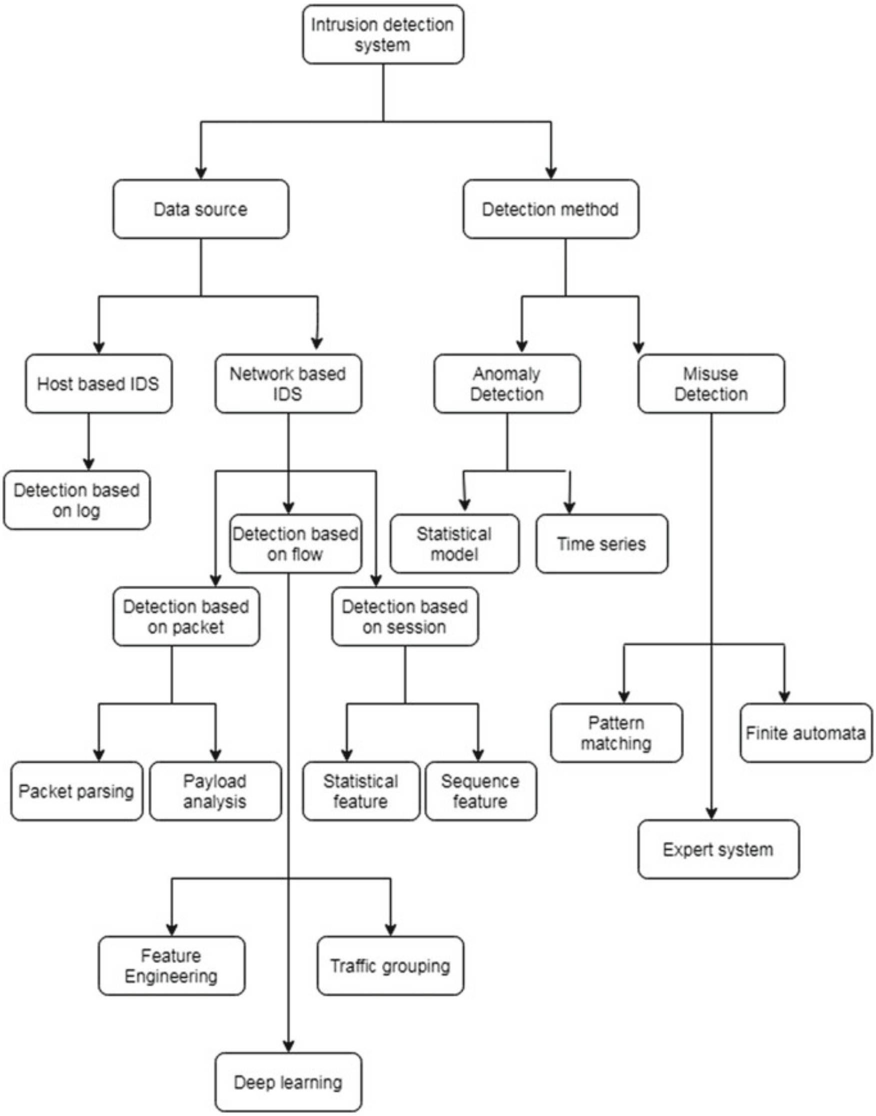


Fig. 1 Detection methodology

The machine learning approach gives the way for implementation of anomaly detection approach due to its various real-time model building approach, and it is gaining popularity in implementation. However, feature selection is also a very important factor for look into. Taking all scenarios into consideration and by looking at the vitality of feature selection, for the implementation of new detection system. This paper is structured as follows: Sect. 1 gives the introduction and various types of detection systems using ML. Section 2 provides a thorough review of various data set. Section 3 illustrates different kinds of machine learning algorithms that have been used over the years. Section 4 gives the summary of various research studies over the years. Section 5 describes various challenges in future research, and in the last Sect. 6 paper is concluded.

2 A Critical Review of Used Data Set

Attacks are changing with time, and hence, there is a need for an updated data set by modifying the old one. This section will discuss the data set used over the years for the implementation of the model as well as some new data sets like CIC-IDS-2017 along with CSE-CIC-IDS-2018 are in use nowadays.

1. **DARPA**: DARPA is the first used data set developed by the MIT Lincoln Laboratory [5]. It is a base data set in the domain of IDS using machine learning. It has 41 features. It does not depict the real network traffic, the absence of false positive, and irregularity in attack data instances are the major flaw of this data set, four categories of attack have been classified PROBE, DOS, U2R, and R2L.
2. **KDD CUP 99**: This is the next version of DARPA developed by the University of California having 41 features in the data set; it contains some redundant and duplicate data samples. The attack type is the same as the previous data set.
3. **NSL-KDD**: This data set is formed by modification in KDD CUP 99 data set, and it consists of the limited number of the attack type.
4. **CAIDA**: Having 20 features, it consists of instances that are specific to Internet activity along with a particular kind of attack.
5. **LBNL**: This data set has 100 h of activity consisting of a packet header for identification of malware.
6. **KYOTO**: It is developed by Kyoto university [20], Having 24 features, it has normal packets and attack-type packets.

For the development of a data set that meets the current technology, some characteristics have been extracted. These characteristics are as follows: available protocol, metadata, anonymity, attack diversity, capturing the complete network traffic, complete network configuration, labeled data, feature set, complete network interaction, and heterogeneity. The CIC-IDS-2017 and CSE-CIC-IDS-2018 data sets are developed by taking into consideration these features [21].

3 Machine Learning Algorithms in IDS

Machine learning is broadly classified into supervised and unsupervised. Supervised counts on the significant information in labeled data, lack of labeled data is a limitation for this method. On the other hand, unsupervised learning extracts valuable insights from unlabelled data, although the detection accuracy of supervised is superior to the unsupervised. Some frequent machine learning algorithms used in IDS are artificial neural network, k-nearest neighbor, naive bayes, support vector, decision tree, and k-means.

Some classifiers are weak in performance and do not perform as expected, and hence, better approach comes into the frame by joining weak classifiers, which gives far better results than earlier; this approach is called the ensembling of classifiers. Ensemble method trained various classifiers and then by voting final output is selected. On the other hand, hybrid methods are organized in a multistage model wherein each stage a classifier is used. The ensemble as well as hybrid classifiers perform better than that of single classifiers, and the main stratagem is which classifier is to be combined and how they should be combined.

4 Summary of Various Research Studies

In this section, application of various machine learning algorithm for the implementation of IDS using different data set has been shown in Table 1. There are various types of attack types whose pattern is unique. Hence, based on the attack characteristics the selection of the data source is important. For example, in the case of a DOS attack, the main feature is to bombarding of packets within a very short period therefore flow data is suitable for detection.

5 Challenges and Future Directions

Models using the machine learning approach have made a huge contribution in the field of intrusion detection systems still some flaws could be rectified.

1. The existed detection system based on feature engineering has high detection accuracy along with a high false alarm rate which is undesirable. A combination of weak classifiers might be a feasible solution to this problem.
2. The diversity of the Internet environment increases the data sets shortage. The novelty of attacks is increasing, and some existing data sets are not good for these kinds of new attacks. Hence, data set alteration in such a way that it should include novel attacks, common attacks, and correspond to the current network environment. Moreover, the available data sets should be properly balanced, less redundant, free of noise, and representative.
3. Poor detection accuracy in the actual environment is still a problem for existing models. Machine learning models have a large ability to detect intrusion but they do not perform well on unknown data; the majority of studies were performed on labeled data. Consequently, performance in the actual environment depends on the coverage of real-world samples.

Table 1 Summary of various research papers

Research paper	Detection	Algorithm	Performance metric	Results
[19]	Intrusion detection	SVM, and chisqselector. Scala programming was done using MLlib	AUPR, AUROC	Low false positive rate, high performanmce
[6]	DDoS attack	K-means, random forest SVM, Java SDK 1.8, matlab 9.1.	F-measures	Automated FFNN edge over all other algorithms
[17]	DoS, R2L, U2R	Decision tree, Naive bayes, SVM and j48	Precision, true positive rate, false positive rate	j48 outperformed all other algorithms
[7]	Intrusion detection	Naive bayes, PART, adaptive boost, and ensemble method. R-programming	Accuracy, recall precision	Ensemble method using bootstrapping outperformed other classifier
[1]	Intrusion detection	PCA-LDA ensemble classification	Accuracy, false positive false negative	Ensemble method using LDA-PCA is outperformed single classifier
[10]	DoS, DDoS attack	SVM, simulation tools	Accuracy, true positive false positive, false detection rate	Light weight IDS for IOT. Packet arrival rate at SVM is good enough
[9]	Anomaly detection	Deep belief network, stacked auto-encoder,	Precision, recall, F1-score	With the use of two level deep learning algorithm achieves accuracy of anomaly detection
[3]	Anomaly detection	PSO, FLN	Detection rate	Better accuracy than the other models based on FLN
[16]	Intrusion detection	SVM, K-means, parsing tool BRO	Precision	HTTP-99.6%, TCP-92.9%, Wiki-99%, Twitter-96%, and Email-93%
[8]	Anomaly detection	Hybrid method using SVM, decision tree and naive bayes	Accuracy, false alarm rate	Accuracy-99.62% and false alarm rate-1.58%
[13]	Intrusion detection	SVM, KNN and PCA.	Accuracy	Accuracy-90.07%
[15]	Intrusion detection	KNN, multilayer perceptron decision tree, naive bayes and SVM	Accuracy, precision F-score	Outcomes shows that decision tree is best classifier for IDS

4. Majority of the studies focused on detection results, and hence, they deploy the complex model and extensive data preprocessing techniques which resulted in lower efficiency.

6 Conclusion

With the advent of Ml, new ideas for IDS have been implemented by various researcher and different forms of classification model have been developed. In this paper, we almost provide the review of numerous papers suggesting various methods for the implementation of IDS, and the primary focus is on anomaly detection.

Various machine learning models were implemented for developing an IDS. Various classifiers have been used with all the possible combinations of feature selection algorithms. It has been found that the methods using deep learning are getting more attention from researchers. KDD99 data set and NSL-KDD data set are the most widely used data sets in this domain. Ensemble and hybrid classifiers have an edge in performance over single classifiers and thus got a good detection rate and the highest predictive accuracy.

References

1. Aburomman AA, Reaz MBI (2016) Ensemble of binary SVM classifiers based on PCA and LDA feature extraction for intrusion detection. In: 2016 IEEE advanced information management, communicates, electronic and automation control conference (IMCEC), pp 636–640 (cit. on p 6)
2. Alazab A, Hobbs M, Abawajy J, Alazab M (2012) Using feature selection for intrusion detection system. In: 2012 international symposium on communications and information technologies (ISCIT), pp 296–301 (cit. on p 3)
3. Ali MH, Al Mohammed BAD, Ismail A, Zolkipli MF (2018) A new intrusion detection system based on fast learning network and particle swarm optimization. IEEE Access 6:20255–20261 (cit. on p 6)
4. AlYousef MY, Abdelmajeed NT (2019) Dynamically detecting security threats and updating a signature-based intrusion detection system's database. Procedia Comput Sci 159:1507–1516 (cit. on p 2)
5. Babu MC, Senthilkumar K (2021) Machine learning based strategies for secure cloud. Mat Today Proc (cit. on p 3)
6. Elsaeidy A, Munasinghe KS, Sharma D, Jamalipour A (2019) Intrusion detection in smart cities using restricted Boltzmann machines. J Netw Comput Appl 135:76–83 (cit. on p 6)
7. Gautam RKS, Doegar EA (2018) An ensemble approach for intrusion detection system using machine learning algorithms. In: 2018 8th international conference on cloud computing, data science & engineering (Confluence), pp 14–15 (cit. on p 6)
8. Goeschel K (2016) Reducing false positives in intrusion detection systems using data-mining techniques utilizing support vector machines, decision trees, and naive Bayes for off-line analysis in SoutheastCon 2016, pp 1–6 (cit. on p 6)
9. Hasan M, Islam MM, Zarif MII, Hashem M (2019) Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches. IoT 7:100059 (cit. on p 6)
10. Jan SU, Ahmed S, Shakhov V, Koo I (2019) Toward a lightweight intrusion detection system for the internet of things. IEEE Access 7:42450–42471 (cit. on p 6)
11. Kaur S, Singh M (2020) Hybrid intrusion detection and signature generation using deep recurrent neural networks. Neural Comput Appl 32 (cit. on p 2)
12. Khraisat A, Gondal I, Vamplew P, Kamruzzaman J (2019) Survey of intrusion detection systems: techniques, datasets and challenges. Cybersecurity 2:1–22 (cit. on p 2)

13. Kumar I, Mohd N, Bhatt C, Sharma SK (2020) Soft computing: theories and applications, pp 565–577. Springer (cit. on p. 6)
14. Liao H-J, Lin C-HR, Lin Y-C, Tung K-Y (2013) Intrusion detection system: a comprehensive review. J Netw Comput Appl 36:16–24 (cit. on p 2)
15. Manhas J, Kotwal S (2021) Multimedia security, pp 217–237. Springer (cit. on pp 2, 6)
16. Mayhew M, Atighetchi M, Adler A, Greenstadt R (2015) Use of machine learning in big data analytics for insider threat detection in MILCOM 2015—2015 IEEE military communications conference, pp 915–922 (cit. on p 6)
17. Mehmood T, Rais HBM (2016) Machine learning algorithms in context of intrusion detection. In: 2016 3rd international conference on computer and information sciences (ICCOINS), pp 369–373 (cit. on p 6)
18. Modi C et al (2013) A survey of intrusion detection techniques in cloud. J Netw Comput Appl 36:42–57 (cit. on p 2)
19. Othman SM, Ba-Alwi FM, Alsohybe NT, Al-Hashida AY (2018) Intrusion detection model using machine learning algorithm on big data environment. J Big Data 5:1–12 (cit. on p 6)
20. Song J et al (2011) Statistical analysis of honeypot data and building of Kyoto 2006+ dataset for NIDS evaluation. In: Proceedings of the first workshop on building analysis datasets and gathering experience returns for security, pp 29–36 (cit. on p 4)
21. Thakkar A, Lohiya R (2020) A review of the advancement in intrusion detection datasets. Procedia Comput Sci 167:636–645 (cit. on p 4)

Automatic Detection of Online Hate Speech Against Women Using Voting Classifier

F. H. A. Shibly, Uzzal Sharma, and H. M. M. Naleer

Abstract Freedom of expression found on social media has various pros and cons. Gender-Based Violence (GBV) is also a major issue in social media. As a part of GBV, hate speech against women is on the rise on all social media. There are some lapses available in the stand-alone classifiers in detecting such speech, and the performance of ensemble classifiers is much better. Also, many research works have focused on common hate speech datasets. Hate speech against women has been used in very few research activities. But such hate speech is very dangerous. As a result, this research employs, Decision Tree (DT), Logistic Regression (LR), Random Forest (RF), and Long Short-Term Memory (LSTM) to compute metrics and performances and then use those algorithms to create a voting classifier to develop a more accurate model for detecting hate speech against at women. Two phases were used in this study. RF, LR, DT, and LSTM were used as foundation stand-alone classifiers in the first phase of the ensemble procedure. In Phase Two, the weights of the second-level classifier were estimated using first-level classifiers. Hate speech against women was detected using an open-source #MeToo dataset that was utilized for training and testing by the researchers. The dataset is publicly available on GitHub which was uploaded by Nazmus Sakib. This dataset consists of 278,765 #MeToo movement posts on social media. It clearly shows that the proposed voting classifier model has the highest values in all metrics including accuracy (89%). When we check the strongly positive classification, the proposed model has performed well in precision (0.90), recall (0.91), and F-measures (0.90) and it can calculate strong positive hate speech more

F. H. A. Shibly (✉)
Assam Don Bosco University, South Eastern University of Sri Lanka, Oluvil, Sri Lanka
e-mail: shiblyfh@seu.ac.lk

U. Sharma
Department of Computer Applications, School of Technology, Assam Don Bosco University, Guwahati, India
e-mail: uzzal.sharma@dbuniversity.ac.in

H. M. M. Naleer
Department of Computer Science, Faculty of Applied Sciences, South Eastern University of Sri Lanka, Oluvil, Sri Lanka
e-mail: drnaleer@seu.ac.lk

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_62

efficiently than other stand-alone classifiers. This voting model takes more time to train since it has multiple models inside. By training it for more epochs, we can further increase accuracy.

Keywords Gender-based violence · Hate speech · Women · Machine learning · Deep learning · And voting classifier

1 Introduction

The contribution of social media in the world of the Internet is becoming day by day more and more vital. Freedom of expression on social media has been accused of being misused by users on several occasions. Various cybercrimes and various controversial issues on social media are taking place daily. Hate speech has become one of the main issues, mainly through social media. Although many research works are ongoing to handle hate speech in social media. But the process of controlling hate speech continues to be limited. Despite numerous research efforts through machine learning, mechanisms are needed to control hate speech.

Hate speech has increased in recent years, both in-person and online. Social media and other Internet platforms are heavily involved in the spawning and spreading of hateful material, ultimately leading to hate crimes [1]. Hate speech can take many forms, including racial discrimination, culture, gender, ethnic group, religious conviction, and some more [2]. GBV is on the rise despite the recent spate of hate speech on social media. We are seeing more hate speech, assault, and cyber-crime targeting women. Now, two of the most vulnerable groups to these kinds of hate crimes are immigrants and women [3]. When hate speech is particularly directed toward women, we refer to it as misogyny [4]. However, it is necessary to control such hate speech against women with proper mechanisms.

A rising number of academics have recently concentrated their efforts on automating the identification of hate words online [5, 6], and several academic events and joint initiatives have centered on this subject [6]. For social media networks, this is considered a difficult task. Major social media sites like Twitter, Instagram, and Facebook, for example, are unable to automatically handle this issue and must rely on their users to report hate speech. Since the situation is so serious and there is so much material being uploaded on the Internet every day, algorithms that can detect hate online comments must be developed [7].

In this paper, researchers apply machine learning algorithms to measure their performances and develop an ensemble method by using the selected machine learning algorithms as a voting classifier to build an efficient method in detecting hate speech against women. A voting classifier is a collection of independent classifiers that are used in conjunction to categorize incoming posts to enhance classification performance [8]. Numerous studies using the ensemble approach have been performed in the general text classification domain and have shown that the ensemble method may significantly improve classification performance [9–11]. Several classifiers that have been used in this ensemble are RF, LR, DT, and LSTM.

2 Related Works

Constitutive nature of speech, and how it shapes social realities, including social and institutional structures are very important in communication process [12]. As women and non-binary individuals engage with the public sphere, they must confront the hegemonic values and norms in the very structures of historical social discourse and the way speech constitutes power. The structures of digital space present, therefore, a paradox for women: They enable agency, but also perpetuate gendered power structures, normalizing sexualized attacks on women as routine expressions of male power and privilege [13, 14].

Whether online or offline, sexist hate speech has significant and pervasive effects on women. They undermine their right to a life free of violence and abuse, whether they be emotional, psychological, or physical. As such, it is also a matter of health and safety. Young women are especially susceptible because of their youth and lack of awareness about available treatments, the latter of which is shared by most of the society [15].

GBV can take many roles, including cyber-related issues and problems and online hate speech directed at women. The terms "cyber violence" and "online hate speech aimed at women" refer to a wide range of cyber violence, including cybers talking, nonconsensual image abuse, and cyber harassment are all examples of this type of behavior, as is the more specific term "sexist hate speech." However, these relatively emerging forms of female violence [16] have no widely acknowledged title.

The United Nations (UN) Committee on the Elimination of Discrimination Against Women (CEDAW) describes GBV as "violence directed against a woman solely for her gender identity or violence that disproportionately affects women," which includes "activities that inflict physical, psychological, or harassments or suffering, as well as threats of such acts, coercion, and other kinds of loss of liberty" [17]; and "actions that cause bodily, mental, or sexual harm or suffering."

The aggregate effect of cyber violence and hate speech directed at women is still being studied, and the data available suggests that these types of violence have the same effect as physical violence directed at women. In the same way that other types of violence against women have immediate and short-term implications, cyber violence and online hate speech also have long-term and intergenerational ramifications for women. Various kinds of violence against women have a detrimental effect on their feeling of security, physical and psychological health, dignity, and legal rights. Furthermore, cyber violence does not need the victim to have experience with it to be effective [18].

So, hate speech against women is a serious issue and the most efficient method for detecting and controlling such online hate crimes is to be controlled without further delay. Meantime, some research works have been carried out to detect and limit hate speech using machine learning and deep learning algorithms. It is well known that Artificial Intelligence (AI) tools and techniques are well enough to handle text classifications.

Automatic hate speech detection was used to examine Indonesian Tweets in [19]. Selected Machine Learning algorithms were used, as well as two assembly methods for balanced and unbalanced datasets: Hard Voting and Soft Voting, respectively.

A hate speech model was created by observing user comments on social media, among other places, to identify hate speech. Logistic Regression, Naive Bayes, Support Vector Machines, XGBoost, and Neural Networks, among other techniques, were used. It has been demonstrated in trials that the XGBoost features exceed the BERT features in terms of performance [20].

An ensemble approach was utilized to enhance the results by combining Logistic Regression and Neural Network models [21]. To detect hate speech aimed toward women and immigrants, an ensemble approach was used. SVM, RF algorithms, and a Bi-LSTM model were utilized to assemble [22]. To train classifiers, bi- and trigram features were utilized. The ensemble technique was used to identify compounds that were intrinsically disagreeable to the human senses, in reference [23]. The binary categorization was accomplished via the usage of Recurrent Neural Networks [23].

Numerous studies have been conducted, with a particular emphasis on GBV. Machine learning techniques were used on Twitter to investigate the conditions surrounding the #MeToo movement (a campaign to eradicate gender-based violence), especially those influencing business and marketing operations, as shown in [24]. In reference [25], the automated identification and classification of misogynistic discourse in Twitter was shown using a variety of supervised classifiers. The feature space of the training dataset was constructed using techniques such as N-grams, linguistic, syntactic, and embedding. One of the primary achievements of this study was to make a corpus of sexist tweets accessible to the academic community.

Based on the literature survey, there are some lapses available in the stand-alone classifiers and the performance of the ensemble classifier is much better. Also, some of the earlier research studies have focused on common hate speech datasets. Hate speech against women has been used in a few studies. But such hate speech is very dangerous. As a result, this research employs RF, LR, DT, and LSTM to compute metrics and performances and then uses those algorithms to create a voting classifier to develop a more accurate model for detecting hate speech against women.

3 Methodology

Two stages were utilized in this study. RF, LR, DT, and LSTM were used as baseline stand-alone classifiers in the first phase. The output of these classifiers served as the foundation for the new feature set used in the second phase. Second-level classifiers were used to estimate the weights of the second-level classifier in phase one. The entire research design is shown in Fig. 1.

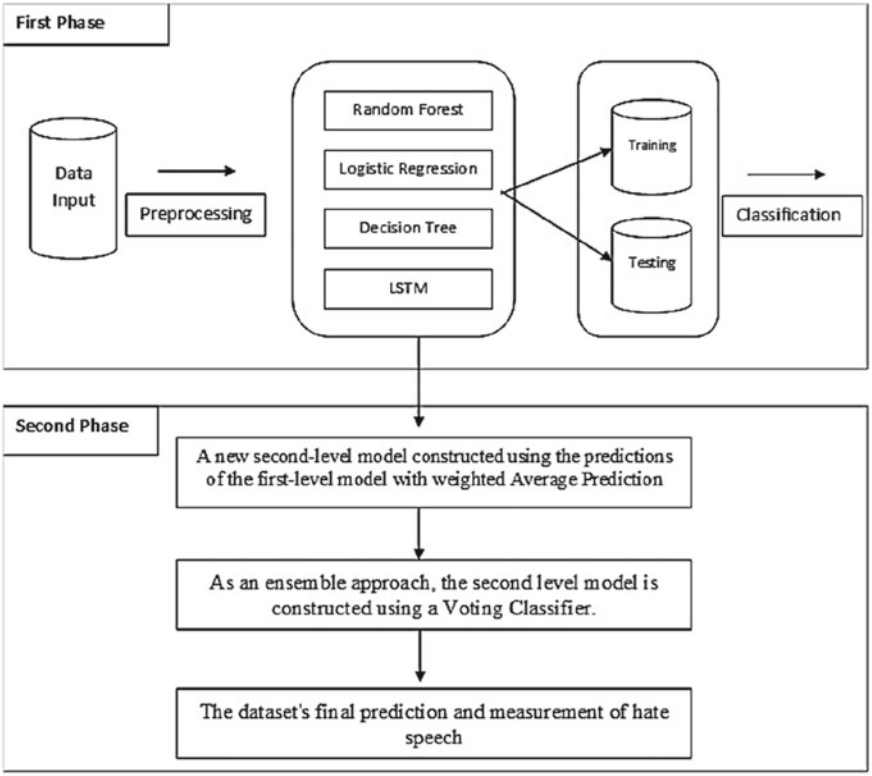


Fig. 1 Proposed research architecture dataset

3.1 Dataset

Researchers have used the open-source #MeToo dataset for training and testing to detect hate speech against women. The dataset is publicly available on GitHub which was uploaded by Nazmus Sakib [26]. This dataset consists of 278,765 #MeToo movement posts on social media. Also, it has four sentiment classifications such as loosely negative, strongly negative, loosely positive, and strongly positive.

3.2 Preprocessing

This part discusses the pre-processing of data utilized in the study. To improve the quality of the #MeToo analysis, postings were pre-processed. The suggested approach makes use of a natural language processing technology. To reduce the size of the text dataset, the pattern of the language of postings was changed to small letters. The test's numerical data was translated to textual representations. Following the elimination of whitespaces, punctuation was deleted to prevent several versions of the same word.

3.3 Proposed Classifiers

This first phase of this research has been applied four algorithms as RF, LR, DT, and LSTM. These algorithms were regarded as foundation learners for the ensemble process.

Random Forest: As the name suggests, the random forest is made up of a varied collection of cooperating decision-making trees [27]. Each tree in the random forest sprays a forecast for a class, and the class that receives the most votes becomes the model's prediction. Individual decision-making trees are created for training purposes by RFs. All the trees' forecasts are merged to produce a single final forecast. They are referred to be ensemble techniques if they produce a final determination based on a collection of data.

Logistic Regression: In the early twentieth century, the biological sciences made use of logistic regression. It was then used for a variety of social science problems. When the dependent variable (target) is categorical, logistic regression is employed [28].

Decision Tree: The Decision Tree technique is a supervised learning methodology that utilizes decision trees to make choices. The decision tree technique, in contrast to other supervised learning algorithms, may be utilized to tackle both regression and classification problems concurrently. It is feasible to build a training model that can correctly predict the class or value of a target variable using a Decision Tree by inferring fundamental decision rules from prior data using this technique (training data) (training data). In Decision Trees, we begin by predicting a class label for a record that is placed at the tree's root. The value of the root property is compared to the value of the record's attribute. Following that, we proceed to the next node [29], which is represented by the branch that corresponds to that value.

LSTM: Long short-term memory is a form of recurrent neural network architecture that is widely applied in deep learning. In contrast to conventional feedforward neural networks, LSTMs feature feedback connections. It is capable of processing not just single data points, but also complete data streams [30].

Voting Classifier: A Voting Classifier is an ML model that learns from an ensemble of models and predicts an output (class) based on the class that has the best chance of being chosen as the output. It simply averages the output classes from each classifier input into Voting Classifier and predicts the output class based on the class with the largest voting majority. Rather than creating and validating separate specialized models, we train a single model on several models and forecast output based on their aggregate majority of votes for each output type [31]. There are two distinct kinds of voting: Hard Voting and Soft Voting. For this research, soft voting has been applied.

4 Findings

The performance of the suggested research technique is evaluated in Table 1. The suggested Voting Classifier is compared to stand-alone classifiers such as RF, LR, DT, and LSTM, and a proposed model is developed utilizing the ensemble technique to create a voting classifier using weighted average metrics.

Accuracy, precision, recall, and F-score are the performance measures computed in the first phase, and the weighted average of those four models have been calculated in the second phase. The voting classifier averages the other four models' prediction and prediction results. So, the voting classifier takes all the other four models' predictions and outputs weighted average predictions. Since the dataset has four classifications, all four have been analyzed over to the five algorithms to calculate a better detection method. In Table 1, all classifiers' performances have been recorded. In Table 1, LS denotes Loosely Negative, LP denotes Loosely Positive, SN denotes Strongly Negative, and SP denotes Strongly Positive. It clearly shows that the proposed model has the highest values in all matrices including accuracy (89%).

Table 1 Measurements of all classifiers

Algorithms	Parameter evaluation				
	Precision	Recall	F-measure	Accuracy	
RF	LS	0.85	0.85	0.85	
	LP	0.88	0.88	0.88	
	SN	0.91	0.91	0.91	88%
	SP	0.90	0.90	0.90	
LR	LS	0.82	0.82	0.82	
	LP	0.85	0.81	0.83	
	SN	0.89	0.90	0.89	86%
	SP	0.87	0.89	0.88	
DT	LS	0.85	0.87	0.86	
	LP	0.88	0.87	0.87	
	SN	0.91	0.91	0.91	88%
	SP	0.90	0.90	0.90	
LSTM	LS	0.83	0.87	0.85	
	LP	0.87	0.86	0.87	
	SN	0.92	0.90	0.91	88%
	SP	0.90	0.89	0.90	
Proposed model (voting classifier)	LS	0.87	0.86	0.86	
	LP	0.88	0.88	0.88	
	SN	0.91	0.92	0.91	89%
	SP	0.90	0.91	0.90	

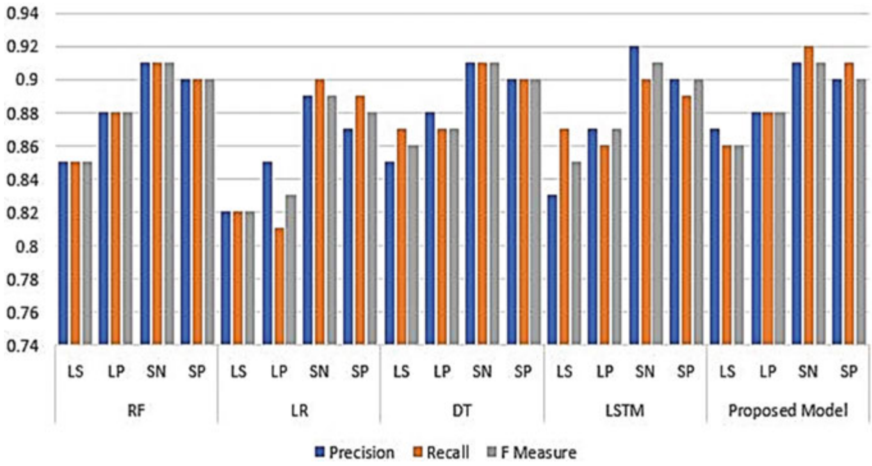


Fig. 2 Matrices of classifiers in all classifications

When we check the SP, the proposed model has performed well in precision (0.90), recall (0.91), and F-measures (0.90) and it can calculate strong positive hate speech more efficiently than other stand-alone classifiers (Fig. 2).

According to the suggested model's confusion matrix (Fig. 3), the model correctly identified the number of positive and negative class data points. Additionally, the model mistakenly categorized a small number of negative class data points as belonging to the positive class and a small number of positive class data points as belonging to the negative class. It demonstrates that the suggested methodology is effective at detecting hate speech against women.

5 Conclusion

The ensemble algorithm's primary objective is to integrate many stand-alone classifiers to improve prediction accuracy. In this study, we integrated stand-alone classifiers, notably RF, LR, DT, and LSTM, to create a voting classifier model for detecting gender-based hate speech, specifically against women, in #MeToo social media postings. When compared to individual base classifiers, this resulted in higher prediction accuracy. This work proposes areas for further research, such as adopting strategies, to increase model prediction scalability in terms of different levels of imbalance in the presence of hate speech phrases in a dataset. Moreover, more machine and deep learning algorithms can be combined as a voting classifier and build a few more efficient models to detect hate content against women. Also, this voting model takes more time to train, since it has multiple models inside. By training it for more epochs, we can further increase accuracy.

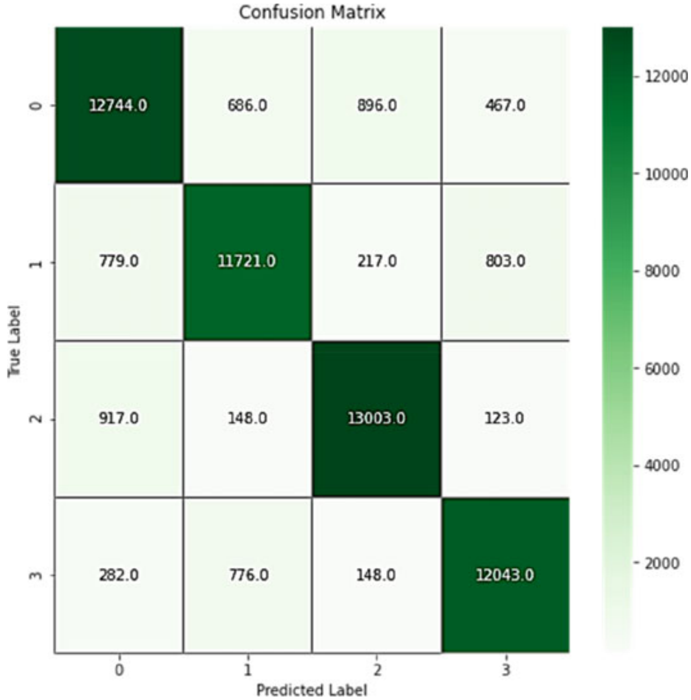


Fig. 3 Confusion matrix of the proposed model

6 Limitations and Future Scope

This research used only the available dataset on the hate speech against women on #MeToo movement posts in social media. There are many posts available against women, but we were able to apply the above dataset only. Also, the researchers used three ML algorithms and one DL algorithm to train and test the detection. There are some more algorithms that will perform well, but for the time being, only specific algorithms were applied. In the future, more ML and DL algorithms can be trained and tested over multilingual datasets to propose a more accurate model.

References

1. Abro S, Shaikh ZS, Khan S, Mujtaba G, Khand ZH (2020) Automatic hate speech detection using machine learning: a comparative study. Mach Learn 10(6)
2. Warner W, Hirschberg J (2012) Detecting hate speech on the world wide web. In: Proceedings of the second workshop on language in social media, 7 Jun 2012, Association for Computational Linguistics, pp 19–26
3. Waseem Z, Hovy D (2016) Hateful symbols or hateful people? Predictive features for hate

speech detection on Twitter. In: Proceedings of the NAACL student research workshop, pp 88–93
4. Manne K (2017) Down girl: the logic of misogyny. Oxford University Press
5. Fortuna P, S´ergio Nunes. (2018) A survey on automatic detection of hate speech in text. ACM Comput Surv (CSUR) 51(4):85
6. Fersini E, Rosso P, Anzovino M (2018b) Overview of the task on automatic misogyny identification at interval 2018
7. Del Arco FMP, Molina-González MD, Martín-Valdivia MT, Lopez LAU (2019) SINAI at SemEval-2019 Task 5: ensemble learning to detect hate speech against immigrants and women in English and Spanish tweets. In: Proceedings of the 13th international workshop on semantic evaluation, pp 476–479
8. Roli F (2015) Multiple classifier systems. Encycl Biomet 1142–7
9. Adeva JG, Atxa JP, Carrillo MU, Zengotitabengoa EA (2014) Automatic text classification to support systematic reviews in medicine. Expert Syst Appl 41(4):1498–1508
10. Dong YS, Han KS (2004) A comparison of several ensemble methods for text categorization. In: Proceedings of the 2004 IEEE international conference on services computing (SCC), 15 Sep 2004. IEEE, pp 419–422
11. Larkey LS, Croft WB (1996) Combining classifiers in text categorization. In: Proceedings of the 19th annual international ACM SIGIR conference on research and development in information retrieval, 18 Aug 1996, ACM, pp 289–297
12. Matsuda M (1989) Public response to racist speech: considering the victim's story. Michigan Law Rev, Legal Storytelling (Aug 1989) 87(8):2320–2381
13. Gurumurthy A, Chami N, Dasarathy A (2021) Recognize, resist, remedy: addressing gender-based hate speech in the online public sphere | IT for change. https://itforchange.net/online-gender-based-hate-speech-women-girls-recognise-resist-remedy
14. Valente M, Martins F, Péricles A, Tavares C, Borges E, Becari J, Pereira C (2020) Discurso de ódio. InternetLab. https://www.internetlab.org.br/pt/projetos/discurso-de-odio/
15. European Youth Centre, Strasbourg (2016) Seminar combating sexist hate speech. Council of Europe
16. Wilk AVD (2018) Cyber violence and hate speech online against women: women's rights and gender equality
17. CEDAW (1992) General Recommendation No. 19 (11th session, 1992). Available at http://www.un.org/womenwatch/daw/cedaw/recommendations/index.html
18. Pew Research Center (2017) Online Harassment 2017. Available at http://assets.pewresearch.org/wpcontent/uploads/sites/14/2017/07/10151519/PI_2017.07.11_Online-Harassment_FINAL.pdf
19. Fauzi MA, Yuniarti A (2018) Ensemble method for indonesian twitter hate speech detection. Indonesian J Electr Eng Comput Sci 11(1):294–299
20. Salminen J, Hopf M, Chowdhury SA, Jung S, Almerekhi H, Jansen BJ (2020) Developing an online hate classifier for multiple social media platforms. HCIS 10(1):1–34
21. Gao L, Huang R (2017) Detecting on-line hate speech using context-aware models. In: Proceedings of the international conference on recent advances in natural language processing, Varna, Bulgaria, pp 260–266
22. Aria N, Vermeer F, Wiltvank G, Goot R (2019) Sthruggle at SemEval-2019 task 5: an ensemble approach to hate speech detection. In: Proceedigns of the 13th international workshop on semantic evaluation, Minneapolis, USA, pp 484–488
23. Pitsilis GK, Ramampiaro H, Langseth H (2018) Effective hate-speech detection in twitter data using recurrent neural networks. Appl Intell 48(12):4730–4742
24. Reyes-Menendez A, Saura JR, Ferrõ F (2020) Marketing challenges in the #MeToo era: gaining business insights using exploratory sentiment analysis. Heliyon 6:e03626
25. Anzovino M, Fersini E, Rosso P (2018) Automatic identification and classification of misogynistic language on Twitter. In: Silberztein M, Atigui F, Kornyshova E, Métais E, Meziane F (eds) Natural language processing and information systems. Springer International Publishing, Cham, Switzerland, pp 57–64

26. Sakib N (2020) GitHub—nsakib1017/ML_Project: here i have used different ML models such as LSTM (2 and 4 layers) CNN (2 and 3 layers) to train it so that it can classify hate speech against women. We have used some open source #metoo datasets for our training. https://github.com/nsakib1017/ML_Project#readme
27. Primartha R, Tama BA (2017) Anomaly detection using random forest: a performance revisited. In: Proceedings of the international conference on data and software engineering, Palembang, Indonesia, pp 1–6
28. Swaminathan S (2019) Logistic regression—detailed overview—towards data science. Medium. https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc
29. Chauhan NS (n.d.) Decision tree algorithm, explained. KDnuggets. Retrieved 8 Sept 2021 from https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html
30. Fernandez AGMLS, Bunke RBH, Schmiduber J (2009) A novel connectionist system for improved unconstrained handwriting recognition. IEEE Trans Patt Anal Mach Intell 31(5)
31. Voting Classifier using Sklearn (n.d.) Prutor online academy (Developed at IIT Kanpur). Retrieved 8 Sept 2021 from https://prutor.ai/voting-classifier-using-sklearn

A Review on EEG Data Classification Methods for Brain–Computer Interface

Vaibhav Jadhav, Namita Tiwari, and Meenu Chawla

Abstract Electroencephalography (EEG) is a technique to quantitatively measure brain activity with high temporal resolution. EEG converts brain activity to time series data with amplitude on the y-axis, and this data can then be used to understand brain functions. Mathematical tools can be applied to this data to extract features and to discriminate them in several classes. Once EEG data is recorded, it is needed to make sense of that data. In the past couple of decades, EEG data has revolutionised the healthcare industry and brain–computer interface (BCI) systems. This is made possible by continuous improvements in EEG data classification methods, which includes improvements in feature extraction and classification algorithms. In this study, methods to classify EEG data for various applications such as medical diagnostics, BCI and emotion detection are reviewed.

1 Introduction

1.1 Electroencephalography (EEG)

Electroencephalography (EEG) quantitatively measures brain activity and converts brain activity to time series data with amplitude on the y-axis, and this data can then be used to understand brain functions. There are other alternatives to EEG for quantizing brain neural activity such as popular magnetic resonance imaging (MRI) and magnetoencephalography (MEG) methods. High temporal resolution (high sampling rate), portability (wearable headsets) and low cost make EEG the perfect choice for brain–computer interface (BCI) applications among other brain data collection tech-

V. Jadhav (✉) · N. Tiwari · M. Chawla
Maulana Azad National Institute of Technology, Bhopal, India
e-mail: vsjadhav.98@gmail.com

N. Tiwari
e-mail: namita_tiwari21@rediffmail.com

M. Chawla
e-mail: chawlam@manit.ac.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_63

niques. , we reviewed methods to classify EEG data for various applications such as medical diagnostics, BCI and emotion detection to name a few.
Working Principle of EEG In EEG, electrodes are placed on the scalp (called non-invasive) or surgically implanted inside the brain (called invasive). These electrodes can detect electric fields caused by the firing of a group of neurons inside of the brain. The firing of neurons in brain happens in two parts. First is action potentials. Action potentials are flow of current in a neuron from one end to another. But, their magnitude is very low and hard to measure. The brain activity that is measured is generated from the second way of conduction in neurons, which is called post-synaptic potentials. These potentials are generated at synapses, where two neural cells communicate with each other. Electric fields generated by these potentials can be detected from the scalp, when thousands of such potentials are generated at a time. These fields induce potential on electrode placed over scalp which is referenced with some reference electrode to get potential difference as output.

1.2 Applications of EEG Data Classification

This section gives answer to question, why to classify EEG data. Following are few of the application of EEG data classification, the positive impact that EEG classification methods can be understood by following applications.

1.2.1 Brain–Computer Interface (BCI)

Brain–Computer Interface(BCI) is a system that takes brain activity (such as EEG), as input and decodes it, in order to communicate with external devices (such as computer). The fact that thoughts can be decoded by external device opens new possibilities for interface between humans and machines. Examples of BCI are:

- MOTOR IMAGERY (MI): It is a system that understands or decodes brain's motor or muscle movement (e.g. hand and legs movement) commands and can make external devices to do intended tasks such as robotic arm movement or mechanical exoskeleton movement with direct control of brain. This can especially benefit paralysed persons as they can use mechanical arms or similar devices in similar way as a healthy individual uses his/her muscles.
- SPELLER: Spelling a word or complete sentences that a person is thinking of via BCI is also topic of many researches in last decade [14, 22, 24]. A unique way to communicate for healthy persons and a necessary tool for paralysed persons with speaking-related complications.
- USER INTERFACES: UI such as cursor movement based on thoughts or applications in game playing to make game playing experience realistic.

MI and SPELLER-related classification methods found in literature are included in this review.

1.2.2 Medical Diagnostics

Brain medical diagnostics is possible with EEG data collected from patients. As classifier algorithms can detect unhealthy and healthy person's EEG data. A lot of literature is available and still it is one of the most found topic in EEG data classification research. In this review, medical diagnostics studies such as [3, 25, 33] are studied.

1.2.3 Emotion Recognition

In last decade, a lot of research has been done on emotional state (valence, arousal, etc.) and mental state (focused, relaxed, etc.) recognition through EEG data. Various methods used in EEG-based emotion recognition are also reviewed in this study.

1.2.4 Sleep Stage Detection

Sleep stages such as rem and deep sleep can be detected though EEG data gathered during sleep. This can also be used in treatment of some medical conditions such as insomnia.

1.2.5 Brain-to-Brain Communication

Once we have BCI, it is natural to think about computer–brain interface (CBI), where machines can give input to brains. Application of CBI can be of type, giving or enhancing vision of persons with medical problems with eye through external vision device like camera feeding input to visual cortex, mimicking eyes. Also, BCI and CBI together can make brain-to-brain communication possible, where system similar to BCI spellers will decode thoughts and then those messages can be transmitted to receiver CBI system, where those messages are again encoded in thoughts and fed to receiver brain. One of the studies related to this is [26].

2 Preprocessing and EEG Artefact Removal

EEG data collection devices are very sensitive to noise, and generally have low signal-to-noise ratio. This is why there are many studies in literature dedicated to artefact removal and preprocessing of EEG.

2.1 EEG Artefacts Removal

Artefacts in EEG are generally noise caused by capture of muscle movements by EEG electrodes. Muscle movements such as eye blink, heart beats, tongue movement and hand movements also generate electric potential field which are much higher in amplitude than those generated in brain, thus EEG electrodes easily pick these potentials and this noise is mixed with brain EEG data. There are various solutions found in literature for removing artefacts such as, using other electrodes devoted specifically to capture muscle movements like eloctroocculograph (EOG) for monitoring eye movement and electrocardiograph (ECG) for making track of heart beats. Then one can epoch or locate the events of muscle movement in time and can process those time windows for EEG electrodes. Other methods involve thresholding or applying band pass filters.

2.2 Feature Extraction and Dimensionality Reduction

A lot of study can be found in literature about feature extraction of EEG. Extracting meaningful and discriminative features can greatly improve classification accuracy. Wavelet transforms, Fourier transform, common spatial patterns, power spectral density, info. gain and CNNs are some of the popular feature extraction methods found in literature. Also principal component analysis, independent component analysis and linear discriminant analysis, are popular for dimensionality reduction.

3 Classification Algorithms

Once raw EEG data is processed, classifier algorithms are applied, and machine learning (ML) models are trained on EEG data in order to get desired classification task done. ML classifiers can be very broadly classified as in [12], as:

- Feature-based methods: In this, handcrafted features are used or manually feature extraction algorithms are selected, and then these features are fed to ML classifiers, e.g. support vector machines (SVMs).
- End-to-End methods: In these methods, ML algorithms learn features as parameters in their training process, e.g. convolutional neural networks (CNNs).

End-to-end ML methods are quite popular among researchers as domain expertise (e.g. medical data diagnosis) is not required for these methods, and ML algorithms does the work of feature selection and extraction as training parameter. This convenience is at the expense of understanding of features used for classification, but a lot of research is being done on getting insights of what representations ML models learns, a study known as explainable artificial intelligence (AO). In Sect. 3, EEG classification methods found in literature are reviewed.

4 Related Work

In this section, review and comparison of different EEG classification method found in literature is done. Also two tables (Tables 1, 2) are included in this review which further describes and compares literature studied. Table 1 gives insights to various public dataset and some private datasets found in literature, datasets are grouped according to their application (data type column in Table 1) in Table 1. Table 2 compares different EEG classification methods on basis of feature extraction methods used, classification algorithms used and achieved accuracy on respective dataset.

4.1 Literature Review

In study [29], Bonn university database is used for binary classification (epileptic seizure or not). They used discrete wavelet transform (DWT) for feature extraction and explored three methods for dimensionality reduction (PCA, ICA and LDA) and used SVM classifier. They found that LDA worked best for dimensionality reduction and reported accuracy 100%. PCA and ICA methods fetched accuracies 98.75% and 99.5%, respectively.

Comparison of different signal decomposition methods for MI-BCI was done in [15]. Here, authors used multiscale PCA to denoise the data (BCIC III- 4a dataset), and then three different signal decomposition methods were compared with kNN classifier. Also authors used only those channel's data which are related to specific intended MI task. It was found that wavelet packet decomposition gave highest average accuracy of 92.8%.

A creative use EEG classification was done in [16]. Authors in this paper experimented to control a quadcopter drone with mental thoughts, one of the best control system possible. Task was to distinguish eight commands for quadcopter control. They used a hybrid EEG-FNIRS (functional near infrared spectroscopy) data acquisition system to reduce complexity of eight class classification to two four class classifiers using FNIRS and EEG data, respectively. Details of eight classes are in Table 1 for corresponding ref [16]. Time window for commands was set to be 2 s for FNIRS and 1sec for EEG. Different band pass cutoffs were applied for EEG and FNIRS. Average accuracy with LDA classifier was 75.6% for FNIRS four class and 86% for EEG four class classification.

Filter bank common spatial patterns are popular method for EEG-based MI classification tasks which uses spectral power modulations as feature. In study [28], authors developed novel architecture of CNN and compared results with FBCSP on BCIC IV-2a and high gamma dataset. They found CNNs outperformed FBCSP with FBCSP accuracy being 82.1%, while that of CNN was 84%. This paper also teaches how to train CNNs for EEG, and visualise what features network has learned.

In [32], researchers have used LSTM to classify MI tasks (BCIC IV-2a dataset) with novel one-dimensional aggregate approximation (1D-AX) to extract effective signal representation for robust classification. Also, inspired by common spatial

Table 1 Datasets found in literature survey

References	Dataset	Data type[a]	Number of classes	Class description	Number of channels	Number of subjects	Sampling frequency (Hz)
[7]	BCI competition III-dataset 4a	Motor imagery	2	• Right hand • Foot	118	5	1000
[16]	Private-used in [16]	Motor imagery and imagination	8	• Mental counting • Mental arithmetic • Mental rotation • Word formation • Two eye blinks • Three eye blinks • Up/down eye movement • Left/right eye movement	16 (FNIRS), 14 (EEG)	10	15.625 (FNIRS), 128 (EEG)
[30]	BCI competition IV-dataset 2a	Motor imagery	4	• Left hand • Right hand • Both foot • Tongue	22	9	250
[30]	BCI competition IV-dataset 2b	Motor imagery	2	• Left hand • Right hand	3	5	250
[19]	BCI competition II-dataset 3	Motor imagery	2	• Left hand • Right hand	3	1	128
[30]	BCI competition IV-dataset 1	Motor imagery	2	• Left, right hand • Foot	59	7	1000
[9]	GigaDB	Motor imagery	2	• Left hand • Right hand	64	52	512
[28]	High gamma	Motor imagery	3	• Left hand • Right hand • Both feet	128	14	512
[24]	private-used in [24]	Speller	5	• Virtual keyboard BCI having 5 classes to select from at a time	1	8	128

(continued)

Table 1 (continued)

References	Dataset	Data type[a]	Number of classes	Class description	Number of channels	Number of subjects	Sampling frequency (Hz)
[22]	private-used in [22]	Speller	50	• 50 words to select from	128 invasive(16 * 8 grid)	1	70–150
[14]	private-used in [14]	Speller	-	• Spell a word[b]	12	20	256
[6]	private-used in [6]	Mental state	2	• Focused • Relaxed	4	5	200
[27]	private-used in [27]	Mental state	2	• Quick math solving • Relaxed	1	8	512
[11]	private-used in [11]	Mental state	2	• Attention • Relaxed	4	120	256
[4]	private-used in [4]	Emotional state	3	• Positive • Neutral • Negative	4	2	150
[20]	private-used in [20]	Emotional state	2	• Like brand-product combination • Dislike it	64	24	1000
[17]	Deap	Emotional state	3	• Arousal • Valance • Dominance	32	32	512
[10]	MindBigData	Imagination	10	• Imagining numbers from 0 to 9	• 1 (MW) • 14 (Epoch) • 4 (Muse) • 5 (Insight)	1	• 512 • 128 • 220 • 128
[3]	Bonn university database	Medical diagnostics	5	• Set A,B,C,D and E (seizure, normal)	128	• 5 healthy • Rest epileptic	173.61
[25]	TUH database	Medical diagnostics	2	• Abnormal • Normal	24 to 36	10874*	250
[33]	Xinjiang medical university hospital data	Medical diagnostics	3	• Preictal • Ictal (seizure) • Interictal	22	13	500

[a] data type based on application of data
[b] for more information on classes to spell a word, refer [14]
* data still increasing in size

patterns (CSP), they used channel weighting to further increase accuracy. Average accuracy for six different binary tasks for nine subjects was 75.283%. Also, average inter-subject transfer learning accuracy (train on eight subjects and with same parameters of neural network test for remaining one subject) of 70.8% was achieved.

P300 spellers are based on steady-state visually evoked potentials (SSVEP), which seems very revolutionary for persons with medical abnormality about speaking to communicate. Authors of [24] developed P300-based speller with custom virtual keyboard with five word classes to select from at a time to spell a word. They used two types of frequency-based features (refer [24]) and a CNN classifier and achieved accuracy of 99.2%. In online task of spelling a word ‘SPELLER’, they achieved accuracy of 97.4% with avg. transfer rate of 49 bpm.

In [27], private dataset was prepared for mental state recognition having two classes, namely focused (quick solve math problem) and relaxed state. Various feature extraction techniques such as power spectral density, spectral centriod, standard division and entropy were explored along with classification algorithms such as LDA, SVM and KNN. All three classifiers used gave best accuracy with power spectral density features with accuracies 95%, 100%, 95% for LDA, SVM and KNN, respectively.

In [14], a P300-based speller was adopted to enable disabled peoples to speak through EEG-based BCI. Study was conducted for twenty participants with amyotrophic lateral sclerosis (ALS), who were not able to talk because of aforementioned complication. A P300 speller was used for performing three spelling tasks:

- block1—spell from already chosen words by participant with cues and feedback on screen
- block2—spell from already chosen words by participant without cues and feedback on screen
- Block3—free use, spell any word

To detect presence of P300 response in data gathered in [14], spatial filter (X-DAWN) was used. LDA was used to classify. For Block1, task accuracy (avg. of all participants) of 97.5% was achieved with more than 90% accuracy for all the participants and 60% participant achieved 100% accuracy. For Block2, task accuracy(avg. of all participants) of 87.25% was achieved with more than 95% accuracy for 65% of the participants. For Block3, task accuracy (avg. of all participants) of 80.205% was achieved.

In [33], seizure detection is done using three-dimensional CNN, on Bonn university database for three classes (interictal, preictal and ictal). Time series EEG was converted to 2D images, and such 2D images generated for each channel was combined into 3D image according to mutual correlation between channels. This 3D input was passed to a 3D CNN, where group normalisation was applied on CNN layers and dropout in fully connected layers. This 3D CNN method was compared with 2D CNN and SVM and found that 3D CNN gave highest accuracy of 92.37% followed by 91.25% of DWT+SVM and then 89.91% of 2D CNN.

In [34], emotion recognition-based on EEG was done through parallel CNN-RNN architecture. Study was performed on Deap dataset which has four classes, where each emotion class was rated on 0–9 scale. In this study, they converted four class problem to two different binary classification problems by choosing five as threshold on original 0–9 scale for low and high for valance and arousal emotions. They got accuracies 90.8% and 91.03% for valence and arousal classification, respectively.

In [6], binary classification for mental state (focused vs. relaxed) was done with various feature extraction and classification methods and their comparison was done. From their original private dataset, five datasets were generated for five feature extraction methods, which were oneR, information gain, correlation, symmetrical uncertainty and evolutionary algorithm. For classification, Naive Bayes, Bayes net, J48, random tree, random forest, MLP and SVM were studied. Highest accuracy of 87.16% was observed on oneR dataset with Random Forrset.

In [18], authors have used new predictor named OPTICAL (which is combination of common spatial patterns (CSP) and LSTM) to classify BCIC IV-dataset 1 and GigaDB dataset. Instead of using LSTM as classifier, here authors have used regression-based output of LSTM as one of the features to train on along with features obtained from CSP + LDA. These features are fed to SVM to classify. They reported improvement of 3.09% and 2.07% accuracy on BCIC IV-1 and GigaDB datasets, respectively.

In [5], authors used evolutionary approach to set hyper parameters of MLP. Also they did adaptive boosting to further increase accuracy. LSTM is compared with deep evolutionary (DEvo) MLP and found that DEvo outperforms LSTM on all three datasets used (mental state [6], emotional state [4], MindBigData [10]). Adaptive boosted DEvo MLP got accuracy of near 84, 97 and 31% on three datasets.

In study [4], emotion sentiment classification is done based on EEG. A private dataset is used for this purpose on which four different feature extraction algorithms were applied to create four datasets from original source dataset. MLP and ensemble random forrest was used to classify, who gave accuracies of 97.89% and 94.89%, respectively with info gain dataset. Info gain was found to be good for feature extraction among compared ones.

In [2], fusion of different CNNs was done with the intuition that different architecture CNNs learn different features. Two novel CNNs, MCNN (fusion of CNNs + MLP) and CCNN (CNN-based AE crossencoding) were trained on BCIC IV-2a and high gamma dataset. MCNN got accuracy of 75.7% and 95.4% on BCIC and high gamma dataset, respectively and those of CCNN were 73.8% and 93.2%.

In [11], a method for doing inter-subject transfer learning was developed, based on CNN for BCI. Private dataset used was having two classes for mental state (attention vs relaxed). Three different types of input data representations (DR1, DR2 and DR3) was obtained from dataset with three different preprocessing techniques. For few shot learning, training on half of the samples of new subject was done to tune network, and 79.26% accuracy was obtained on DR1. Also 76.68% accuracy was obtained in zero shot on DR3.

In [8], a novel joint method is proposed to classify MindBigData (insight version) and compared with FFT joint and SVM joint methods. This novel joint method is comprised of FFT + PCA + weighted KNN. Novel joint method outperformed other two methods with accuracy of 84%, while FFT joint and SVM joint gave 79% and 70%, respectively.

In [19], motor imagery classification is done on BCIC datasets. They propose continuous wavelet transform (CWT) for feature extraction and 2D CNN for classification. Proposed method was compared with short time Fourier transform (STFT) + 1D CNN. CWT was done for three different mother wavelets, which were Morlet, mexican hat and bump. Maximum accuracy of 92.9% was given by bump base CWT + 2D CNN on mu + beta frequency band.

In [23], LSTM and novel improved neural network (INN) are applied on Bonn university database to classify normal and abnormal brain conditions. INN is a CNN with novelty lying in one dimensional gradient descent activation function with radial basis operations in initial layers of INN. Authors of this paper also trained other classifiers like SVM, logistic regression, etc. and found that INN outperforms others with accuracy of 78.92%.

ML task can be very broadly classified in two categories as feature-based (handcrafted features) and end-to-end (features are learned). End-to-end methods such as CNN are very popular because it does not require domain expertise, thus handcrafted ML methods are comparatively less explored. Thus, in [12], researchers explored and compared feature-based methods with end-to-end approach, with hypothesis that given equal opportunity to feature-based methods they can perform as good as end-to-end methods. They found classification accuracies on TUH abnormal dataset (pathological versus non-pathological) for both the methods to be in narrow range between 81 and 86%. Even if hand crafted feature based ML techniques reach near end-to-end methods, still both methods have their advantages and disadvantages.

In [35], a novel neural network was proposed for seizure detection on Bonn university database. They used three layer CNN and three layer fully connected layers all using dropout. Original Bonn database has five classes, in this study, different combinations of those five classes were used to train for several two class and three class classifications. Full five class classification was also done with accuracy of 93.55%. Different combinations of two class gave accuracy in range 97.63–99.52% and that for three class was 96.73–98.06%.

In [20], a novel RNN-based method is used to classify whether customer will like some brand and product combination or not on private dataset. Here, t-distributed stochastic neighbour embedding (t-SNE) method is used to extract features and those features then fed to RNN. This t-SNE + RNN approach was compared with t-SNE + SVM, t-SNE + MLP, PCA + RNN, ICA + RNN, LSTM, CNN classifiers and found that t-SNE + RNN produces highest accuracy among them which was 87.37%.

In [22], a speller was developed for a paralysed personnel which can classify 50 words (classes) that the subject was thinking of custom DL model was used to classify and to further increase accuracy and to output meaningful sentences, the output of custom DL model was fed to NLP model. Use of NLP model reduced word error rate by 35% compared to original. Average accuracy of 47.1% was achieved for 50 class classification which is equivalent to 97.8% of binary class accuracy.

Table 2 Comparison of classification methods found in survey

Ref. year	Dataset	Data type	Accuracy	Feature extraction methods	Classification algorithm
[29] (2010)	Bonn university database	Medical diagnostics	• 98.75 (DWT + PCA + SVM) • 99.5 (DWT + ICA + SVM) • 100 (DWT + LDA + SVM)	• Discrete wavelet transform (DWT)	• SVM
[15] (2017)	BCI competition III-dataset 4a	MI	• 92.8 by multiscale PCA + Wavelet packet decomposition	• Empirical mode decomposition • Discrete wavelet transform • Wavelet packet decomposition	• K-nearest neighbour
[16] (2017)	Private dataset used in [16]	MI, imagination	• 75.6 with FNIRS • 86 with EEG	• Signal peak • Signal mean • Signal minimum	• LDA
[28] (2017)	• BCI competition IV-dataset 2a • High gamma	MI	• 82.1 with FBCSP • 84 with deep CNN	• Spectral power modulations for FBCSP • Learned in training for CNN	• Filter bank common spatial pattern (FBCSP) • CNN
[32] (2018)	BCI competition III-dataset 4a	MI	• 75.28 Avg. • 70.8 Avg. with zero shot transfer learning	• One-dimensional aggregate approximation • Channel weighting	• LSTM
[24] (2018)	Private dataset used in [24]	Speller	• 99.2 • 97.4 real time(online task)	• Two types of features based on frequency domain (FFT based)	• CNN
[27] (2018)	Private dataset used in [27]	Mental state	• 95 (PSD + LDA) • 100 (PSD + SVM) • 95 (PSD + KNN)	• Power spectral density (PSD) • Spectral centroid • Entropy	• LDA • SVM • KNN
[14] (2018)	Private dataset used in [14]	Speller	• 97.5 (avg. for task1) • 87.25 (avg. for task2) • 80.205 (avg. for task3)	• Spatial filter (X-DAWN)	• LDA
[33] (2018)	Xinjiang medical univ. hospital database	Medical diagnostics	• 92.3 (3D CNN) • 89.91 (2D CNN) • 91.25(entropy + DWT + SVM)	• Learnt in training by CNN • DWT	• 3D CNN • 2D CNN • SVM
[34] (2018)	Deap	Emotional state	• 90.8 (low or high valance) • 91.03 (low or high arousal)	• Learnt in training by CN	• CNN and RNN in parallel
[6] (2018)	Private- used in [6]	Mental state	• 87.16 (oneR + RF)	• oneR • Information gain • Symmetrical uncertainity • Evolutionary algorithm	• Random forest • MLP • SVM • Naive bayes

(continued)

Table 2 (continued)

Ref. year	Dataset	Data type	Accuracy	Feature extraction methods	Classification algorithm
[18] (2019)	• BCI competition IV-dataset 1 • GigaDB	MI	• 78.33	• CSP + LDA • CSP + LSTM-regression	• SVM
[5] (2019)	• Dataset used in [6] • Dataset used in [4] • MindBig Data	Mental state, emotional state, imagination	• 84 (dataset[6]) • 97 (dataset[4]) • 31.35 (MindBigData)	• Features extracted as in [4, 6] • Features from evolutionary algo.	• MLP • LSTM
[4] (2019)	Private dataset used in [4]	Emotional state	• 97.89 Ensemble (info gain + RF) • 94.89 (info gain + MLP)	• OneR • Bayes network • Info gain • Symmetric uncertainty	• Ensemble-RF • MLP
[2] (2019)	• BCI competition IV-dataset 2a • High gamma	MI	• 75.7 (MCNN, BCIC IV) • 95.4 (MCNN, high gamma) • 73.8 (CCNN, BCIC IV) • 93.2(CCNN, high gamma)	• feature fusion of learnt features in CNN training	• MCNN(CNN + MLP) • CCNN (CNN based on autoencoder crossencoding)
[11] (2019)	Private-used in [11]	Mental state	• 79.26 (few shot) • 76.68 (zero shot)	• Learnt features in CNN training	• CNN
[8] (2019)	MindBig data (insight)	imagination	• 84 (novel joint method) • 79 (FFT joint method) • 70 (SVM joint method)	• FFT + PCA	• Weighted KNN
[19] (2019)	• BCIC IV-dataset 2b • BCIC II-dataset 3	MI	• 92.9 (CWT + 2D CNN) • 89.3 (STFT + 1D CNN)	• CWT • STFT	• 2D CNN • 1D CNN
[23] (2020)	Bonn university database	Medical diagnostics	• 78.92 novel INN • 71.38 (LSTM)	• Learned in training	• INN (CNN with novel act. fn.) • LSTM
[12] (2020)	TUH abnormal EEG corpus v2.0.0	Medical diagnostics	• 86.57 temporal CNN • 85.87 hand crafted features + riemannian geometry	• Handcrafted • Learned by CNN	• TCN (temporal CNN) • Riemannian geometry • Random forest
[35] (2020)	Bonn univ. database	Medical diagnostics	• 97.63–99.52 (2 class tasks) • 96.73–98.06 (3 class tasks) • 93.55 (5 class)	• Learned in training by CNN	• CNN
[20] (2021)	Private-used in [20]	Emotional state	• 87.37	• t-SNE	• RNN
[22] (2021)	Private-used in [22]	Medical diagnostics	• 47.1 (50 class classification)	• Handcrafted	• DL model + NLP model

Some other good reviews found were [1, 13, 21, 31]. In [1], deep learing(DL) methods for MI EEG were reviewed and in [13] a comparison of DL in EEG for last ten years literature is studied.

5 Conclusion and Future Work

In this study different methods to classify EEG data were reviewed for various applications like MI-BCI, BCI-speller, medical diagnostics and emotion recognition. EEG classification methods have exponentially improved in last decades. Although, it was observed that very few studies are getting conducted on more than ten class classification. To make EEG-based BCI as a part of human life, it is important that future studies to be done should be inclined towards multiclass EEG classifications with more than ten classes, such multiclass system will be a great start for adopting EEG-based BCI as part of general purpose devices.

References

1. Al-Saegh A, Dawwd SA, Abdul-Jabbar JM (2021) Deep learning for motor imagery EEG-based classification: a review. Biomed Signal Process Control 63:102172 (cit. on p 10)
2. Amin SU, Alsulaiman M, Muhammad G, Mekhtiche MA, Hossain MS (2019) Deep Learning for EEG motor imagery classification based on multilayer CNNs feature fusion. Future Gen Comput Syst 101:542–554 (cit. on pp 9, 12)
3. Andrzejak RG et al (2001) Indications of nonlinear deterministic and finitedimensional structures in time series of brain electrical activity: dependence on recording region and brain state. Phys Rev E 64:061907 (cit. on pp 3, 7)
4. Bird JJ, Ekart A, Buckingham CD, Faria DR (2019) Mental emotional sentiment classification with an eeg-based brain-machine interface. In: Proceedings of the international conference on digital image and signal processing (DISP ’19) (cit. on pp 6, 9, 11, 12)
5. Bird JJ, Faria DR, Manso LJ, Ekárt A, Buckingham CD (2019) A deep evolutionary approach to bioinspired classifier optimisation for brain-machine interaction. Complexity (cit. on pp 9, 11)
6. Bird JJ, Manso LJ, Ribeiro EP, Ekárt A, Faria DR (2018) A study on mental state classification using EEG-based brain-machine interface. In: 2018 international conference on intelligent systems (IS), pp 795–800 (cit. on pp 6, 9, 11)
7. Blankertz B et al (2006) The BCI competition III: validating alternative approaches to actual BCI problems. IEEE Trans Neural Syst Rehabil Eng 14:153–159 (cit. on p 6)
8. Chen D et al (2019) Novel joint algorithm based on EEG in complex scenarios. Comput Assisted Surg 24:17–125 (cit. on pp 9, 12)
9. Cho H, Ahn M, Ahn S, Kwon M, Jun SC (2017) EEG datasets for motor imagery brain-computer interface. GigaScience 6, gix034 (cit. on p 6)
10. David V (2015) MindBigData: the “MNIST” of brain digits http://www.mindbigdata.com/opendb/. Accessed 7 Aug 2021 (cit. on pp 7, 9)
11. Fahimi F et al (2019) Inter-subject transfer learning with an end-to-end deep convolutional neural network for EEG-based BCI. J Neural Eng 16:026007 (cit. on pp 6, 9, 12)
12. Gemein LA et al (2020) Machine-learning-based diagnostics of EEG pathology. NeuroImage 220:117021 (cit. on pp 4, 10, 12)

13. Gong S, Xing K, Cichocki A, Li J (2020) Deep learning in EEG: advance of the last ten-year critical period (cit. on p 10). arXiv:2011.11128
14. Guy V et al (2018) Brain computer interface with the P300 speller: usability for disabled people with amyotrophic lateral sclerosis. Annals Phys Rehabil Med 61:5–11 (cit. on pp 3, 6, 8, 11)
15. Kevric J, Subasi A (2017) Comparison of signal decomposition methods in classification of EEG signals for motor-imagery BCI system. Biomed Signal Process Control 31:398–406 (cit. on pp 5, 11)
16. Khan MJ, Hong K-S (2017) Hybrid EEG-fNIRS-based eight-command decoding for BCI: application to quadcopter control. Frontiers Neurorobot 11, 6 (cit. on pp 5–7, 11)
17. Koelstra S et al (2011) Deap: a database for emotion analysis; using physiological signals. IEEE Trans Affect Comput 3:18–31 (cit. on p 7)
18. Kumar S, Sharma A, Tsunoda T (2019) Brain wave classification using long short-term memory network based OPTICAL predictor. Sci Rep 9:1–13 (cit. on pp 9, 11)
19. Lee HK, Choi Y-S (2019) Application of continuous wavelet transform and convolutional neural network in decoding motor imagery brain-computer interface. Entropy 21:1199 (cit. on pp 6, 10, 12)
20. Ma Q, Wang M, Hu L, Zhang L, Hua Z (2021) A novel recurrent neural network to classify EEG signals for customers' decision-making behavior prediction in brand extension scenario. Front Human Neurosci 15:54 (cit. on pp 7, 10, 12)
21. Minguillon J, Lopez-Gordo MA, Pelayo F (2017) Trends in EEG-BCI for daily-life: requirements for artifact removal. Biomed Signal Process Control 31:407–418 (cit. on p 10)
22. Moses DA et al (2021) Neuroprosthesis for decoding speech in a paralyzed person with anarthria. New England J Med 385:217–227 (cit. on pp 3, 6, 10, 12)
23. Nagabushanam P, Thomas George S, Radha S (2020) EEG signal classification using LSTM and improved neural network algorithms. Soft Comput 24:9981–10003 (cit. on pp 10, 12)
24. Nguyen T-H, Chung W-Y (2018) A single-channel SSVEP-based BCI speller using deep learning. IEEE Access 7:752–1763 (cit. on pp 3, 6, 8, 11)
25. Obeid I, Picone J (2016) The temple university hospital EEG data corpus. Front Neurosci 10:196 (cit. on pp 3, 7)
26. Rao RP et al (2014) A direct brain-to-brain interface in humans. PloS One 9:e111332 (cit. on p 3)
27. Rashid M, Sulaiman N, Mustafa M, Khatun S, Bari BS (2018) The classification of EEG signal using different machine learning techniques for BCI application. In: International conference on robot intelligence technology and applications, pp 207–221 (cit. on pp 6, 8, 11)
28. Schirrmeister RT et al (2017) Deep learning with convolutional neural networks for EEG decoding and visualization. Human Brain Mapp 38:5391–5420 (cit. on pp 6, 7, 11)
29. Subasi A, Gursoy MI (2010) EEG signal classification using PCA, ICA, LDA and support vector machines. Expert Syst Appl 37:8659–8666 (cit. on pp 5, 11)
30. Tangermann M et al (2012) Review of the BCI competition IV. Front Neurosci 6:55 (cit. on p 6)
31. Torres EP, Torres EA, Hernaández-Álvarez M, Yoo SG (2020) EEG-based BCI emotion recognition: a survey. Sensors 20:5083 (cit. on p 10)
32. Wang P, Jiang A, Liu X, Shang J, Zhang L (2018) LSTM-based EEG classification in motor imagery tasks. IEEE Trans Neural Syst Rehabil Eng 26:2086–2095 (cit. on pp 7, 11)
33. Wei X, Zhou L, Chen Z, Zhang L, Zhou Y (2018) Automatic seizure detection using three-dimensional CNN based on multi-channel EEG. BMC Med Inform Decis Making 18:71–80 (cit. on pp 3, 7, 8, 11)
34. Yang Y, Wu Q, Qiu M, Wang Y, Chen X (2018) Emotion recognition from multi-channel EEG through parallel convolutional recurrent neural network. In: 2018 international joint conference on neural networks (IJCNN), pp 1–7 (cit. on pp 8, 11)
35. Zhao W et al (2020) A novel deep neural network for robust detection of seizures using EEG signals. Comput Math Methods Med (cit. on pp 10, 12)

A Variational Autoencoder—General Adversarial Networks (VAE-GAN) Based Model for Ligand Designing

K. Mukesh, Srisurya Ippatapu Venkata, Spandana Chereddy, E. Anbazhagan, and I. R. Oviya

Abstract COVID-19 pandemic has disrupted the normal functioning of the world at both the physical as well as biological level. Various biotechnological approaches form the core for finding the drug for COVID. But at the backend, it is highly supported by intensive computational algorithms for drug designing approaches. Artificial intelligence is one such area that can be used to mimic the living system by training it with multi-dimensional datasets. The rapid advancements in the field of artificial intelligence and machine learning have facilitated to build high-accurate generative models. Deep learning is a subset of artificial intelligence that is being used in the present study for ligand designing models. The results obtained by a deep learning-based model called variational autoencoder—general adversarial networks (VAE-GAN) show promising results in terms of ligand design and can also be utilized for drug repurposing. In this paper, we have attempted to build a VAE-GAN model which was trained on isomeric simplified molecular-input line-entry system (Iso-SMILES) data and molecular structure images of the COVID-19 drug dataset. The Iso-SMILES and molecular structure data analysis are useful, but the system needs more improvement to cater to the data loss in terms of spatial structures and orientation of the chemical compounds taken for the analysis. However, the system can be

K. Mukesh · S. Ippatapu Venkata · S. Chereddy · E. Anbazhagan
Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Chennai, India
e-mail: ch.en.u4aie20039@ch.students.amrita.edu

S. Ippatapu Venkata
e-mail: ch.en.u4aie20010@ch.students.amrita.edu

S. Chereddy
e-mail: ch.en.u4aie20024@ch.students.amrita.edu

E. Anbazhagan
e-mail: ch.en.u4aie20001@ch.students.amrita.edu

I. R. Oviya (✉)
Department of Sciences, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Chennai, India
e-mail: ir_oviya@ch.amrita.edu

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_64

optimized with the help of higher hardware support and increased training datasets, which can help in more precise analysis for generating ligand molecules of interest.

Keywords COVID-19 · Deep learning · Generative adversarial networks · Generative models · Variational autoencoders

1 Introduction

The recent pandemic of COVID-19 causes severe illness by primarily affecting the respiratory system and is caused by SARS-CoV-2. The originally realized case was distinguished in Wuhan, China, in December 2019. The infection has since spread around the world, prompting a severe pandemic. During this pandemic, many attempts were made, and many techniques were developed all over the world to discover the drugs which are effective against COVID-19. Many algorithms which are developed during this pandemic for drug discovery include deep learning, machine learning, and reinforcement algorithms. The role of machine learning and deep learning algorithms has a huge impact on the current drug discovery/repurposing process. With the help of AI algorithms, we can narrow down the compounds which are more relevant and perform better in clinical trials.

A major role of these computational algorithms is in the quantitative structure–activity relationship also called QSAR methods [1] which mainly relies on basic principles of biological activity of a ligand and its functional association with the molecular structure. Optimization of molecular structure to address chemical absorption, distribution, metabolism, excretion, and toxicity (ADMET) properties are highly relevant with the use of AI techniques.

De novo drug design is a computation method for creating new chemical structures from atomic components. Structure-based and ligand-based design are two common approaches that rely on the features of a biological target's active site. Artificial intelligence, which includes machine learning, is a new field that has had a positive impact on medication development. Artificial neural networks and reinforcement learning architectures are combined in deep reinforcement learning, a subset of machine learning.

1.1 Motivation

Using deep generative models for ligand designing introduces a novel and promising solution to drug discovery. Drug discovery is an expensive and extensively time-consuming process. In traditional methods of drug discovery, it takes around 10–12 years with a lot of human effort and costs a huge amount. But the new deep generative models made this process faster and cost-effective to discover a new drug and aids with the traditional drug discovery process. VAE-GAN being one of the high-performance deep generative models can be a viable option for ligand designing, thus making the process of drug repurposing time and cost-effective. In this paper, we are training VAE-GAN models with two different input data, one being Iso-SMILES data and another one is molecular structure image samples.

1.2 Organization

The paper is organized into five sections. The first section being introduction and Sect. 2 consists of literature review and discusses about the previous research done in developing deep generative models for generating new drugs. Section 3 describes about the data preprocessing stage, including the selection of desired properties from the huge dataset and the conversion of data to proper format, so that the model can be trained. Section 4 covers the implementation of VAE-GAN model and its architecture. Section 5 discusses about the results obtained from the model along with the incurring losses of each layer of the VAE-GAN model. Finally, Sect. 6 concludes the paper and discusses about the challenges encountered in development and training of the model.

2 Literature Review

Li et al. [2] proposed a quantum GAN deep generative model with a hybrid generator to discover new drug molecules. The QGAN model takes less training time compared to the classical GAN models but requires high-performance quantum computers and higher degree hardware support. A CycleGAN-based model named Mol-CycleGAN [3] was successfully applied to generate compounds with more similarity in structure to the real compounds. Another CycleGAN-based model named LA-CycleGAN was proposed by Feng Wang et al. [4] to generate molecules by embedding the long short-term memory and attention mechanism in the model. It showed highly accurate results when compared to the Mol-CycleGAN.

The goal of drug design is to create chemical molecules with specified desirable qualities. D4GAN [5], a new drug design approach that can generate molecular samples that fit a specific set of desirable characteristics. To directly deal with molecules recorded as text sequences, they utilized principles from boundary-seeking GAN (BSGAN) and objective-reinforced GAN (ORGAN). Furthermore, it has been combined with a variational autoencoder for the GAN generator to improve the stability and quality of sample model creation while avoiding mode collapse.

The generative topographic mapping (GTM) [6] method can be used to explore the latent space of SMILES-based autoencoders and build targeted chemical libraries.

They created a sequence-to-sequence neural network with bidirectional long short-term memory layers and trained it on ChEMBL23's SMILES strings. The reconstruction rates of the test set molecules were extremely high (>98%), equivalent to those reported in related works. They used GTM to show the autoencoder latent space on a two-dimensional topographic map. By sampling associated latent space points and decoding them to SMILES, targeted map zones can be used to generate unique chemical structures.

Recent advances in deep learning-based generative models may provide a feasible alternative for more efficient molecular design. Some research works used a variational autoencoder to optimize molecular characteristics in a latent space in which molecules are expressed as a real vector. In addition, the researchers used the adversarial autoencoder (AAE) [7] and Bayesian optimization to create ligands for the dopamine type 2 receptor. They examined the VAE and AAE molecular generation models in terms of reconstruction error and variability of the output molecular fingerprints.

3 Dataset Analysis and Preprocessing

The data of drugs/chemical compounds that are related to COVID-19 was collected from the PubChem database. A total of 1617 compounds were available in the PubChem database. The data collected from PubChem must be preprocessed to get the model to work on the desired attributes from the dataset. The dataset contains all the information regarding each compound. Among these, we were interested only in the Iso-SMILES data and the molecular structure images. The Iso-SMILES data is then preprocessed and given as input for training the VAE-GAN model. Also, the structural images of the molecules were used to train another model with the same VAE-GAN architecture.

The initial set extracted from dataset was the compound name and the Iso-SMILES string for the first model, and the Iso-SMILES data is replaced with the compound's molecular structure image in the latter model. All the characters present in the SMILES string have been extracted and are replaced with a one-hot encoded values to convert from characters to numerical data. Then, the Iso-SMILES strings are padded with a constant value to make all the compounds of same length. For the second model, the images are converted to grayscale and are reshaped to 200 $\times$ 200 pixels. We have split the dataset into 80% for training model and remaining 20% as test data.

4 Implementation

Variational autoencoder—general adversarial networks (VAE-GAN) [8, 9] is a deep generative model which integrates both VAE and GAN to provide a robust deep

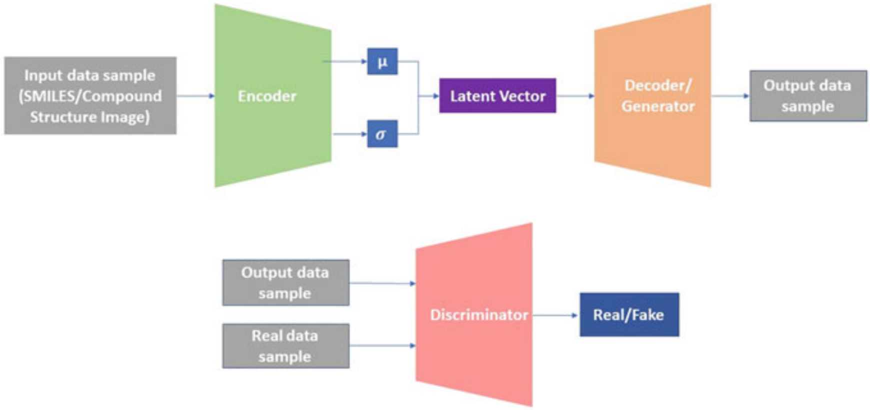


Fig. 1 Functioning of the VAE–GAN model along with its pipeline and data processing stages

learning architecture. VAE describes an observation in latent space in a probabilistic manner [10] which consists of an encoder and a decoder. The encoder transforms the input data into a latent representation, which is a compressed low-dimensional representation of the input data. The decoder reconstructs an approximate representation of the input data from the latent representation. GAN's are a deep generative model used to discover hidden patterns in the data, to generate new examples from the original data which consists of a generator and a discriminator. Generally, in a VAE-GAN model, the decoder in VAE acts as the generator for GAN. The samples that are theoretically possible which are based on the input training set are generated (Fig. 1).

The VAE-GAN model is implemented with three sub-models—an encoder, a decoder, and a discriminator. The encoder has an input layer, followed by two convolutional layers and a dense layer. The input data passes through the encoder and gets reduced to a low-dimension latent space. The decoder has a dense layer followed by three transpose convolutional layers with ReLU activation function. The latent space is then passed through the decoder to generate new examples. Finally, the discriminator has an input layer followed by two convolutional layers and a dense layer. The generated example from decoder passes through the discriminator, which classifies the example as real or fake. The logit and sigmoid functions are used to calculate the loss of each sub-model and visualize it graphically. For the generator, we are calculating the fake loss and for the discriminator, we are calculating both real and fake loss.

5 Results and Discussion

After training the model, the results of our workflow are providing substantial outputs from the training dataset. But still by altering the standard deviations and the learning rates, the model can yield a more precise output than the persisting ones. Due to lack of computational power, we have trained the model on a relatively smaller dataset and for a lower number of epochs. After completion of the training phase, we have been analyzing the error and loss of the model and the molecules it generated.

The output of model trained on images is giving more satisfactory results in generating molecules when compared to the model trained with Iso-SMILES data, and the generated molecules has been verified with the rdkit library. Studies with GAN models have been implemented in biological sciences to scan images as inputs for the identification of the attribute of interest [11, 12]. This gives us more understanding of the usability of the VAE-GAN model in a different area of sciences and our focus on novel ligand modification in this study can be justified. Figure 2 represents the model's discriminator and generator losses, and we can infer that the generator fake loss is high. This can be also observed in the output images shown in Fig. 3.

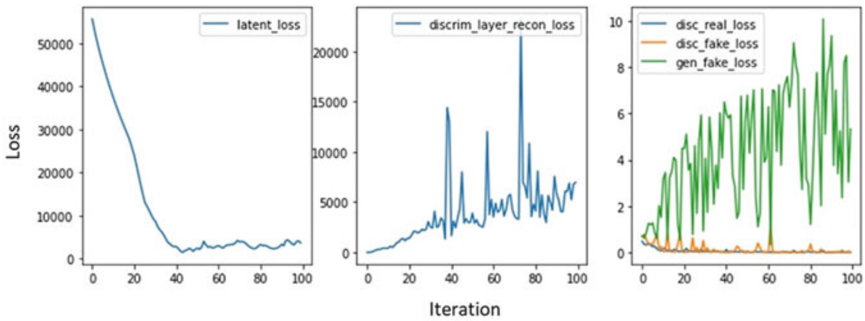


Fig. 2 Training loss of our model is plotted. **a** Represents the latent loss, **b** represents the discriminator layer's reconstruction loss, and **c** demonstrates real and fake data losses after training

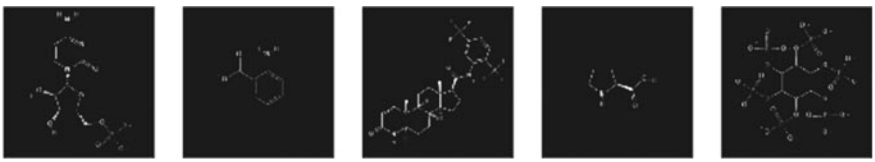


Fig. 3 Representing the output molecules of our model

6 Conclusion and Future Scope

The present model with molecular structure image as input does not show much accuracy but we can understand the structure of the molecules. These models may further be evaluated and used for QSAR studies for novel drug discovery processes. The main challenge is the huge image dataset for which we need heavy computation which increases the chance of improvising the present model by taking many training examples and adjusting the parameters of our GAN model such as learning rate and the standard deviation of the latent points in generator block. The other challenge was both discriminator and generator should perform with equal efficiency for the GAN to function properly or else there is a high chance of collapsing of the model and we must start the process again from the beginning. The application of VAE-GAN in drug designing studies like QSAR also opens a vast opportunity for the generation of novel molecules or to deal with the existing molecules and target the ADMET properties so that the molecules could be processed for further studies. Most of the naturally occurring compounds show a great deal of activity against COVID, but they are not utilized due to their high toxicity rate in living systems. Hence, modifications to decrease the toxicity and increase its potency is the key motive in drug repurposing studies and the present approaches can fill in such lacunae.

References

1. Abdel-Ilah L, Veljović E, Gurbeta L, Badnjević A (2017) Applications of QSAR study in drug design. IJERTV6IS060241
2. Li J, Topaloglu RO, Ghosh S (2021) Quantum generative models for small molecule drug discovery. In: IEEE transactions on quantum engineering, vol 2, pp 1–8, 2021, Art no. 3103308. doi: https://doi.org/10.1109/TQE.2021.3104804
3. Maziarka Ł, Pocha A, Kaczmarczyk J et al (2020) Mol-CycleGAN: a generative model for molecular optimization. J Cheminform 12:2. https://doi.org/10.1186/s13321-019-0404-1
4. Wang F, Feng X, Guo X, Xu L, Xie L, Chang S (2021) Improving de novo molecule generation by embedding LSTM and attention mechanism in CycleGAN. Front Genet 12:709500. https://doi.org/10.3389/fgene.2021.709500
5. Krishnan SR, Bung N, Vangala SR, Srinivasan R, Bulusu G, Roy A (2021) De Novo structure-based drug design using deep learning. J Chem Inf Model. doi: https://doi.org/10.1021/acs.jcim.1c01319. Epub ahead of print. PMID: 34792338
6. Sattarov B, Baskin I, Horvath D, Marcou G, Jannik Bjerrum E, et al (2018) De Novo molecular design by combining deep autoencoder recurrent neural networks with generative topographic mapping. J Chem Inf Model Am Chem Soc 59(3):1182–1196. ff10.1021/acs.jcim.8b00751ff.ffhal-02346951f
7. Makhzani A, Shlens J, Jaitly N, Goodfellow I (2016) Adversarial autoencoders. In: International conference on learning representations, Caribe Hilton, San Juan, Puerto Rico, 2–4 May 2016
8. Larsen ABL, Sonderby SK, Larochelle H, Winther O (2016) Autoencoding beyond pixels using a learned similarity metric. In: International conference of machine learning (ICML)
9. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. In: Advances in neural information processing systems (NIPS), pp 2672–2680. [1406.2661] Generative Adversarial Networks (arxiv.org)

10. Kingma DP, Welling M (2014) Auto-encoding variational Bayes. In: Proceedings of the international conference on learning representations (ICLR), 2014. [1312.6114v10] Auto-Encoding Variational Bayes (arxiv.org)
11. Shashank K, Sajithvariyar VV, Sowmya V, Soman KP, Sivanpillai R, Brown GK (2020) Identifying epiphytes in drones photos with a conditional generative adversarial network (C-GAN). In: The international archives of the photogrammetry, remote sensing and spatial information sciences, ASPRS annual conference virtual technical program, 2020, volume XLIV-M-2–2020. https://doi.org/10.5194/isprs-archives-XLIV-M-2-2020-99-2020
12. Aswin S, Sajithvariyar VV, Sivanpillai R, Vishvanathan S, Brown GK, Shashank A, Soman KP (2021) Effect of annotation and loss function on epiphyte identification using conditional generative adversarial network. In: International conference on advances in electrical, computing, communication and sustainable technologies (ICAECT), pp 1–6. doi:https://doi.org/10.1109/ICAECT49130.2021.9392478

A Brief Review on Protein Classification Based on Functional, Behavioral, and Structural Properties Using Data Mining Techniques

Stuti Majumdar, Suprativ Saha, and Tanmay Bhattacharya

Abstract Knowledge retrieval from a large amount of biological database is one of the challenging tasks, nowadays. Numerous types of data mining techniques are applied to execute the same. For a few years, several researchers have established a lot of information retrieval procedures to extract knowledge from a wide-reaching amount of biological informations like protein and genes. In this paper, the authors try to make a brief review regarding these classification techniques along with their accuracy and computational time, which can classify protein into its family. The authors also try to mention the name of databases and procedures which are used to validate these classification approaches. In the end, a comparative analysis between these classification approaches was established alone with limitations and chance of improvement areas. Finally, a brief idea regarding the protein classification concept along with its need is clearly emphasized here.

Keywords ANN · Fuzzy · Rough set · String kernel · HMM · Di-sulfide bond

1 Introduction

In modern, biology has become an information science. Nowadays, the outstanding rate of genome or protein sequences is saving up as sequencing data. In this framework, the traditional data processing method fails to categorize it. In the twenty-first century, biological data mining is driving the competitive race toward unprecedented biological discovery opportunities. It provides a fruitful technique and consists of biological contents such as DNA, RNA, and protein. Various approaches were introduced to classify protein, involving combination of machine learning, statistics, and

S. Majumdar · S. Saha (✉)
Department of Computer Science and Engineering, Brainware University, Barasat, Kolkata 700125, India
e-mail: reach2suprativ@yahoo.co.in

T. Bhattacharya
Department of Information Technology, Techno India, Saltlake, Kolkata 700091, India

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_65

database system. In this field, machine learning is mainly focused on predicting protein functions from their sequence structure. Within the various classification approaches, feature grouping is the most wanted drawback. This paper depicts the limitation of feature extraction procedures, which are used to classify the protein sequences. In Sect. 2, we have given a short literature survey on several types of protein classification techniques proposed by various researchers invented from time to time followed by a proper comparative analysis in Sect. 3. Finally, Sect. 4 concludes the article.

2 Literature Review

Feature extraction, knowledge generation, and application of soft computing approaches are the three major steps to classify unknown protein to its family. Several researchers had applied these steps in various ways and generated an accurate classifier. Here a brief explanation of these classifiers is below.

2.1 Neural Network-Based Classifier

In the case of protein classification, a neural network-based classifier provides a remarkable achievement based on the accuracy of classification. In paper [1], Jason et al. proposed a neural network-based classifier for classifying unknown proteins by extracting features by using the 2-gram encoding method and 6-letter exchange group method with 90–92% accuracy. To increase the exactness, Wu et al. [2] proposed n-gram encoding method in place of the 2-g encoding method [1] to extract feature values. Here, besides the increment of accuracy, the computational time is also increased. Zainuddin et al. [3] formulated a pattern matrix using 2-g encoding method for reducing the computational time. If the previous matrix would be unable to classify, then the result of the 3-g encoding method would be attached to the pattern matrix and so no. Further for increasing the accuracy of classification, a probabilistic neural network model [4] involving self-organizing map network was proposed by Rao et al. Unfortunately, this technique cannot give impressive results in the case of various unclassified parameters. The problems arise by the previous technique in [4] are overcome by the backpropagation neural network technique in [5]. Here authors used an extreme learning machine to minimize the error of classification which can automatically increase the accuracy level of the classification. After that, Saha et al. [6] proposed the saturation point of n-gram encoding method [2] to minimize the computational time of classification and keep remain the accuracy level. After extracting mean value and the standard deviation by n-gram encoding method, the saturation point was fixed on 5. In [7], Saha et al. proposed the floating mean value policy to extract the standard deviation value of protein using n-gram encoding methods. The accuracy of the classification may be increased some time, and computational time should be decreased in this way.

2.2 Fuzzy Logic-Based Classifier

To overcome the various drawbacks of neural network-based classifier, Mohamed et al. [8] implemented a fuzzy rule-based model using molecular weight, isoelectric point, hydropathy properties as the features for classifying unknown protein with 93% high accuracy. To decrease the computational time, Mansoori et al. [9] wanted to classify protein sequence using feature group distribution policy like small, medium and large, based on fuzzy rule-based algorithm. After that, Mansoori et al. [10] proposed a rank-based algorithm to distribute the feature group, which can reduce the computational complexity, but the accuracy remains same. In [11], Saha et al. applied the positional weighted average molecular weight and positional average isoelectric value to fuzzy ARTMAP model to increase the accuracy. Furthermore, in [12] Saha et al. formed four groups of features with the amalgamation of positional [11] and non-positional weighted values of the features. This research proves that the combination of positional average molecular weight and positional average isoelectric point provides the most valuable result of classification based on the accuracy and computational time of classification.

2.3 Rough Set Classifier

To handle the large amount of data and to identify the necessary features, rough set approach was provided a new classifier by Pawlak [14]. Rahman et al. [13] presented a new classifier involving rough set which can categorize the voluminous protein data based on structural and functional properties of protein with 97.7% accuracy. Later, in [15] Yellasiri et al. used rough set classifier to extract all the necessary features for classification which can achieve a satisfactory exactness level of 97% without increasing the computational time.

2.4 Classifier Based on Hybridize Model

Hybridization approach may be solved the various problems based on accuracy and computational time of protein classification, which are generated by the previous non-hybrid classifier. In paper [16], Sen et al. proposed a python-based standalone tool, i.e., PyPredT6 to predict the T6 effector proteins. The prediction is done by the agreement of multilayer perceptron, SVM, K-nearest neighbor, Naive Bayes, and random forest to extract 175 rare protein from SecReT6 and SecretEPDB databases. Filatov et al. [17] suggest a new convolutional kernel function known as the Lempel–Ziv–Welch (LZW) kernel which is dependent on support vector machine, Fisher Kernel, and hidden Markov model with the accuracy of 73.9%. For prediction of transmembrane beta-barrel proteins, Kazemian et al. [18] proposed a cascading classifier. The algorithms like resilient backpropagation, Fletcher–Powell conjugate gradient, conjugate gradient with Powell–Beale restarts have been used to obtain the exactness of 76.3% with a precision of 0.831 and probability detection of 0.799. After that, Saha et al. [19, 20] proposed a feature grouping hybridization procedure that involves three phases like combination of fuzzy ARTMAP model, neural network system, and rough set classifier. KMP algorithm was applied in this procedure to reduce the computational time with accuracy 91%. Later in paper [21], Saha et al. modified the previous 3-phase model into 4-phase model to increase the accuracy of the classification.

2.5 Classifier Involving Frequency-Based Encoding Method

Frequency-based encoding method was represented by Iqbal et al. [22] for classifying the protein sequence and determine their function and structure. The existence frequency of each amino acid in protein sequences is used as a numeric feature value for encoding the input protein sequences. The effectiveness of this technique was tested using different classifications and learning algorithms, and the result showed that the decision tree classification algorithm based on the criteria of classification specificity, accuracy, F-measure, sensitivity, etc. The accuracy of the classification over the yeast protein sequence data is taken from the UniPortKB database achieves 88.7%.

2.6 Classifier Involving Distance-Based Encoding Method

Besides the behavioral approaches of classification, structural feature extraction procedure can provide valuable techniques in protein classification. In this case, a distance-based encoding method with 91.2% accuracy was proposed by Iqbal et al. [23] where the features are extracted from the input protein sequences and find the distance between the occurrences of the same amino acid which is used as a feature value and tested with different classifiers. Faye et al. [24] developed a computational and statistical method to encode protein sequences based on amino acid decomposition. This approach was used on neural network and support vector machines classifiers with well-known GPCR datasets. Here, neural network-based classifier achieves better performance with an accuracy range of 94–98%.

2.7 Graph Kernel-Based Classifier

Qiangrong et al. [25] used graph kernel-based model combining based on the neural network on protein classification. The adjacency matrix of the graph was taken as the entry point. It was combined with the information of edge and vertexes weights to the adjacency matrix. The database is provided by the PDB, and the protein structure datasets contain the D&D, ENZYMES, NCI1, and NCI109. Accuracy of VES kernel in D&D, ENZYMES, NCI1, and NCI109 is 99.23, 93.50, 99.52, and 99.19% from which they determine that NCI-1 gives the more accurate result which can be improved further.

2.8 Classifier Using Probabilistic Approach

During the time of drug design, the important task is to study and classify the unknown protein into a known protein family. Satpute et al. [26] proposed a probabilistic approach involving feed-forward and feedback ANN, Naive-based, SVM, and decision tree to classify protein with efficiency of 63%, 59%, 68%, and 84%, respectively. In critical cellular processes long non-coding RNAs (lncRNAs) are actively related. These are associated with numerous diseases in the biological experiment. In this paper, Sunil Kumar et al. [27] demonstrate a heterogeneous information network (HIN) to predict lncRNA diseases by using support vector machine classifier. Performance of the model is executed on IncRNA disease database and validated using standard statistical metrics which generated an AUC value of 0.87.

2.9 Tool-Based Classifier

During the case of bacterial identification and bacterial protein detection, MALDI-TOF is a swift-sensitive technique. Tomachewski et al. [28] developed a tool, i.e., Ribopeaks, for the classification of bacteria through m/z data from ribosomal protein with the database of more than 28,500 bacterial taxonomic records. Ribopeak categorizes 111 strains (95.68%) at the species and genus levels. To predict the impact of mutations in protein-protein complexes, Amengual-Rigo et al. [29] were proposed fast classifier and an open source ,UEP, which is a confidential algorithm for figuring constructive and destructive mutations in protein-protein compounds skilled on interactome data. UEP algorithm is based on simple three-body unity capability copied from interactome data.

2.10 Classification Using Deep-Learning

To classify circular RNA from other long non-coding RNA, Chaabane et al. [30] and Benson et al. [31] proposed the ACNN-BLSTM sequence descriptor and reverse complement matching (RCM) descriptor combines the asymmetric convolution neural network (ACNN) with the bidirectional long short-term memory network (BLSTM) where the shared representations across different techniques are unified. Derbel et al. [32] suggested the disease named entity recognition approach using long-short dependencies that are based on two stages deep neural network model to identify the disease name. New POS-based tagging schema has been developed that divides up the dominant class into a smaller and more balanced unit to overcome the problem of an unbalanced dataset given by the BMEWO tagging schema and to enforce sequence modeling. To solve the dimensionality Word2vec, GloVe, and Text model, algorithms are used with combining 85.59% accuracy.

2.11 String Kernel Method and Weighting Scheme-Based Classifier

Identification of protein similarity is an important task for protein sequence classification and homology detection. The string kernel method-based classifier was proposed by Spalding et al. [33] which developed a scheme for efficient estimation of suitable kernel parameter values. Here the Kullback–Leibler (KL) distance was calculated among the observed k-mar frequencies and the theoretical k-mar frequencies of protein data. This classifier reached 87.5% accuracy. Zaki et al. [34] depict a technique to extract feature vectors using hidden Markov model, which was applied to the classifier that was able to train the data in high dimensional spaces. This approach also used support vector machines classified for learning the boundaries between structural protein classes. The performance of this technique was examined in SCOP database with 99.03% accuracy.

2.12 Classifier Involving Fourier Transform

Ali et al. [35] predicted the functional classification of protein sequences based on a set of features involving fast Fourier transformation (FFT) of molecular weight of each protein sequence was applied on SCOP database. The multilayer backpropagation neural network classifier has been used for this classification. When it was applied on restricted level of SCOP, the accuracy level was enhanced up to 96%.

2.13 Classifier Using Feature Hashing Technique

Dimensionality reduction technique based on feature selection is very crucial for the performance and the complexity of the learning algorithms. To execute it, Caragea et al. [36] proposed feature hashing technique, for reducing the complexity of learning algorithm where input belongs to high-dimensional space. Generally, N-gram encoding method produces high-dimensional data for the large value of N. This high-dimensional space was reduced to low-dimensional space by feature hashing mechanism using a hash function. Lastly, this feature hashing classifier was achieved 82.83% accuracy.

2.14 Tree-Based Classifier

Busa-Fekete et al. [37] proposed the phylogenetic analysis approach with 93% accuracy, followed by Tree Insert and TreNN algorithm for protein classification. This representation was also compared with different classification models like BLAST, Smith–Waterman, and local alignment kernel. Boujenfa et al. [38] proposed TreeKNN algorithm that exceeded the performance of TreeNN [37]. This algorithm was performed based on a phylogenetic tree that followed phylogenomic approach, used to overcome the systematic errors related to sequence comparison tools and also to increase the classification performance. The evaluation result of these two algorithms was demonstrated on four datasets such as 3PGK, COG, CATH95, SCOP95, and two alignment programs BLAST and ClustalX.

2.15 Hidden Markov-Based Classifier

Desai [39] had proposed hidden Markov models based classifier with 94% accuracy, which were performed in three phases such as training, decoding, and evaluation to identify functional properties of input data. In the training phase of HMM, a bunch of known sequences was used to learn a model to generate the sequences of trained cluster. In the decoding phase, the most probable path from the trained model was obtained by using the Viterbi algorithm for the given sequence. The last phase calculated the probability of a sequence belonging to the same class as a model by using the Viterbi algorithm.

2.16 Classifier Using GA/SVM

Selecting the most informative features and reducing the dimensionality of the feature vector are an important task in protein sequence classification. Zhao et al. [40] proposed a classifier with the amalgamation of genetic algorithm and support vector machine framework. The GA was allowed to iteratively search for a feature vector that optimized the fitness function. This model was applied on six protein superfamilies which belonged to PIR protein database and classified the protein sequence with 99.24% accuracy.

2.17 Protein Structural Classification

Protein structural classification is another important classification approach to classify protein based on protein chemical structure. Rahman et al. [41] had proposed hierarchy tree structure with six major features of a protein-like sequence comparison, cluster index, connectivity, structure comparison, taxonomic, and interactivity with 98% accuracy. Later, AlQuraishi [42] had established ProteinNet which is a used to evaluate the relationship between protein sequence and structure. The accuracy of the final version of ProteinNET was around 90%. After that, Jain et al.[43] had proposed random forest-based automated structural classifier for small proteins. This approach achieves up to 94% accuracy. Then Arun et al. [44] had coined a protein classifier with 94% accuracy using averaged NMR chemical shifts based on protein residues. BioMagResBank and RefDB were used to obtain the data files. Finally, Mirceva et al. [45] had provided a hidden Markov model for each protein fold class. At the end, proper combination of the di-sulfide bond for classifying unknown protein was identified by Saha et al. [46, 47] which finally used as the structural feature of protein. This feature used in the rule-based classifier to classify protein with above 93% accuracy.

3 Comparative Analysis

See Table 1.

Table 1 A comparative analysis of various classification approaches with their accuracy and chances of improvement

Name of classifier	Accuracy (%)	Drawback or chances of improvement
Classifier based on ANN	94.5	Nonlinearity concept should be included
Classifier based on fuzzy concept	96.2	Storage problem must be avoided and some improvement requires for linear database also
Rough set-based classifier	97.7	More constraints must be included in algorithm beside knowledge base concept
Neuro-fuzzy hybrid model	96.1	Computational time and complexity must be reduced of this problem
Classifier based on frequency-based encoding method	88.7	Dimensionality-related problem should be avoided
Classifier based on distance-based encoding method	98	Non-structural analysis is also required to include
Graph kernel-based classifier	99.52	Accuracy was dropped in case to structural analysis. More concentration is required on it
Classifier using probabilistic approach	87	Require more concentration on the dimension reduction
Tool-based classifier	95.68	Require to reduce computational time
Deep learning-based classifier	85.59	LSTM techniques should be introduced for better accuracy
String kernel and string weighted-based classifier	99	Only applicable on protein sequence, structural analysis is very much required
FFT-based classifier	96	Totally depends on database, general rules should be applicable
Feature hashing-based classifier	83	Dependent classifier. Only provides the good result in case of dimension reduction approach
Tree-based classifier	93	Structural analysis should be incorporated
Hidden Markov-based classifier	94	Beside probabilistic approach, require more concentration on feature extraction techniques
Classifier based on GA and SVM	99.24	In case of structural analysis, accuracy does not remain same. Require a proper solution on it
Structural-based classifier	96.33	Beside di-sulfide bond, other bonding structure regarding AA chain should be incorporated

4 Conclusion

Due to the growing and developing scope of biological data, the widely vast science of data mining within the domain of bioinformatics is seemly perfect. The integration of biological databases is also a difficulty. Because of this huge research area, it is obvious that biological databases depict a huge number of challenges. Data mining and bioinformatics are fast-growing research areas today. So, it is an important research issue in bioinformatics and develops new data mining methods. In the case of protein sequence classification finding accurate classifiers to classify unknown protein sequences into a proper family is an important area. This chapter contains so various types of classification techniques involving data mining for classifying unknown protein sequences. It is a review and comparative study between various protein sequence classifiers. In the conclusion part, it seems that no classification technique achieves the highest level of accuracy with less computational time at the same time. This chapter will help to plan a new model for classifying unknown protein sequences.

References

1. Wang JTL, Ma QH, Shasha D, Wu Cathy H (2000) Application of neural networks to biological data mining: a case study in protein sequence classification. KDD, Boston, MA, USA, pp 305–309
2. Wu C, Michael B, Sailaja S, Jerry M (1995) Neural networks for full-scale protein sequence classification: sequence encoding with singular value decomposition. Kluwer Academic Publishers, Boston. Manufactured in The Netherlands. Mach Learn 21:177–193
3. Zainuddin Z, Kumar M (2008) Radial basic function neural networks in protein sequence classification. Malaysian J Math Sci 2(2):195–204
4. Nageswara Rao PV, Uma Devi T, Kaladhar D, Sridhar G, Rao AA (2009) A probabilistic neural network approach for protein superfamily classification. J Theoret Appl Inf Technol
5. Wang D, Huang G-B (2005) Protein sequence classification using extreme learning machine. In: Proceedings of international joint conference on neural networks (IJCNN2005), Montreal
6. Saha S, Bhattacharya T (2018) A novel approach to find the saturation point of n-gram encoding method for protein sequence classification involving data mining, LNNS Springer, ICICC 2018, pp 101–108
7. Saha S, Bhattacharya T (2019) An approach to find proper execution parameters of n-gram encoding method based on protein sequence classification. CCIS, Springer, vol 1046, ICACDS-2019, Ghaziabad, pp 294–303
8. Mohamed S, Rubin D, Marwala T (2006) Multi-class protein sequence classification using fuzzy ARTMAP. In: IEEE conference, pp 1676–1680
9. Mansoori EG, Zolghadri MJ, Katebi SD, Mohabatkar H, Boostani R, Sadreddini MH (2008) Generating fuzzy rules for protein classification. Iran J Fuzzy Syst 5(2):21–33
10. Mansoori EG, Zolghadri Mansoor J, Katebi SD (2009) Protein superfamily classification using fuzzy rule-based classifier. IEEE Trans Nanobiosci 8(1):92–99
11. Saha S, Bhattacharya T (2020) A new protein sequence classification approach using positional-average values of features. In: Soft computing: theories and applications. Advances in intelligent systems and computing, vol 1053. pp 703–712. Springer

12. Saha S, Bhattacharya T (2020) An approach to select the proper combination within positional and non-positional average values of features. In: Protein classification advances in intelligent systems and computing, vol 1154. Springer, pp 913–924
13. Rahman SA, Bakar AA, Hussein ZAM (2009) Feature selection and classification of protein subfamilies using rough sets. In: International conference on electrical engineering and informatics. Selangor
14. Pawlak Z (2002) Rough set theory and its applications. J Telecommun Inf Technol
15. Yellasiri R, Rao CR (2009) Rough set protein classifier. J Theory Appl Inf Technol
16. Sen R, Nayak L, De RK (2019) A python-based prediction tool for identification of type VI effector proteins JBCB, vol 17, pp 1950019-1–1950019-17
17. Gleb F, Bruno B, Attila K, Farkas SZ (2018) LZW-Kernel: fast kernel utilizing variable length code blocks from LZW compressors for protein sequence classification. IR, pp 3281–3288
18. Kazemian HB, Grimaldi CM (2020) Cascading classifier application for topology prediction of transmembrane beta-barrel proteins. JBCB, pp 1–15
19. Saha S, Chaki R (2013) A brief review of data mining application involving protein sequence classification. AISC, ACITY 2012, vol 177, pp 469–477. Springer, Chennai
20. Saha S, Chaki R (2012) Application of data mining in protein sequence classification. In: IJDMS, vol 4, no 5, AIRCC, pp 103–118. https://doi.org/10.5121/ijdms.2012.4508
21. Saha S, Bhattacharya T (2019) An approach to enhance the design of protein sequence classifier using data mining. Procedia Comput Sci J Elsevier 167C:717–726
22. Iqbal MJ, Faye I, Said AM, Samir BB (2014) An efficient computational intelligence technique for classification of protein sequences. IEEE 2014:1–6
23. Iqbal MJ, Faye I, Said AM, Samir BB (2013) A distance-based feature-encoding technique for protein sequence classification in bioinformatics. CYBERNETICSCOM 2013. IEEE 2013:1–5
24. Iqbal MJ, Faye I, Said AM, Samir Brahim Belhaouari (2016) Classification of GPCRs protein using a statistical encoding method. IEEE 2016:1–5
25. Qiangrong J, Guang Q (2019) Graph kernels combined with the neural network on protein classification. JBCB 17:1950030-1–1950030-11
26. Satpute BS, Yadav R (2019) An efficient machine learning technique for protein classification using probabilistic approach. AISC, vol 828, pp 405–413. Springer
27. Sunil Kumar PV, Gopakumar G (2019) Inferring disease and pathway associations of long non-coding RNAs using heterogeneous information network model. JBCB, vol 17, pp 1950020-1–1950020-18
28. Tomachewski D et al (2018) Ribopeaks: a web tool for bacterial classification through m/z data from ribosomal proteins. Bioinformatics 34(17):3058–3060
29. Amengual-Rigo P et al (2018) UEP: an open-source and fast classifier for predicting the impact of mutations in protein complexes
30. Chaabane M et al (2020) circDeep: deep learning approach for circular RNA classification from other long non-coding RNA. Bioinformatics 36(1):73–80
31. Benson DA et al (2017) GenBank. Nucleic Acids Res 45:D37
32. Derbel H, Chaibi AH, Ghezala HHB (2019) Disease named entity recognition using long-short dependencies, JBCB pp 1–15
33. Spalding JD, Hoyle DC (2005) Accuracy of string kernels for protein sequence classification, ICAPR 2005. Springer (LNCS) vol 3686
34. Zaki NM, Deri S, Illias RM (2005) Protein sequences classification based on string weighting scheme. Int J Comput Internet Manage 13-1:50–60
35. Ali AF, Shawky DM (2010) A novel approach for protein classification using Fourier transform. IJEAS 6(4):2010
36. Caragea C, Silvescu A, Mitra P (2012) Protein sequence classification using feature hashing. Proteome Sci 10(Supple 1):S14
37. Busa-Fekete R, Kocsor A, Pongor S (2010) Tree-based algorithms for protein classification. Int J Comput Sci Eng (IJCSE)
38. Boujenfa K, Essoussi N, Limam M (2011) Tree-kNN: a tree-based algorithm for protein sequence classification. Int J Comput Sci Eng (IJCSE) 3:961–968, ISSN: 0975-3397

39. Desai P (2005) Sequence classification using hidden Markov model
40. Zhao X-M et al (2004) A novel hybrid GA/SVM system for protein sequences classification. In: IDEAL 2004, Springer (LNCS) 3177, pp 11–16
41. Rahman MM, Alam AU, Abdullah-Al-Mamun, Mursalin TE (2011) A more appropriate protein classification using data mining, (JATIT), pp 33–43
42. AlQuraishi M (2019) ProteinNet: a standardized data set for machine learning of protein structure. BMC Bioinform 20, Article number: 311
43. Jain P, Hirst JD (2010) Automatic structure classification of small proteins using random forest. BMC Bioinform 11, Article number: 364
44. Kumar AV, Ali RFM, Cao Y, Krishnan VV (2015) Application of data mining tools for classification of protein structural class from residue based averaged NMR chemical shifts. Biochim Biophys Acta 1854(10 0 0):1545–1552
45. Mirceva G et al (2009) HMM based approach for classifying protein structures. Int J Bio-Sci Bio-Technol 1(1)
46. Saha S, Paul T, Bhattacharya T (2021) A study to find a potent feature by combining the various disulphide bonds of protein using data mining technique. Netw Model Anal Health Inform Bioinform 10(36):1–12
47. Saha S, Sarkar S, Bhattacharya T (2021) A review of protein structure classification along with a proposed classifier using data mining techniques. In: Lecture notes in networks and systems, vol 170, pp 179–188. Springer.

Intelligent System for Bi-Modal Recognition of Apparent Personality Traits (iSMART)

Cdr Devraj Patel and Sunita V. Dhavale

Abstract Personality of an individual has been a promising variable to understand himself and furthermore the others in the society. It is the logical arrangement of an individual's attributes like thoughts, feelings, attitudes, behaviour and capability that makes an individual selective. Our personality likewise influences our decisions, medical conditions, assumptions, inclinations and prerequisites. In the scenario of 4G/5G and COVID pandemic, the majority of individuals are dependent on the web gateways as their essential intuitive vehicle for their own and expert necessities; accordingly, it has been a fundamental significance for us to consequently perceive the personality traits of the individual on the opposite side of the screen. Mental analysts have tracked down that an interaction of just 100 ms is adequate to shape judgement about any individual. Thinking about a similar idea towards execution of profound learning for recognition of personality traits, in this work, we propose an intelligent model (iSMART), a combination of depth-wise separable convolution neural network (2D-CNN) and long short-term memory with attention (LSTMwA), that extracts audio and video features through parallel networks and predicts the ultimate personality score of a person. With the top to bottom trial and error, it has been seen that the depth-wise separable CNN reduces the quantity of trainable parameters without compromising the test precision. It is a compelling and lightweight model for recognition of personality traits utilising bi-modular data sources. It likewise accomplishes better accuracy as compared with the outcomes got by the top scoring teams in the ChaLearn Looking at People challenge ECCV 2016. Our proposed model can possibly empower the system with better psychological understandings and improved human–computer interaction.

C. D. Patel (✉) · S. V. Dhavale
Department of CSE, Defence Institute of Advanced Technology (DIAT), Pune, India
e-mail: devraj_pcse19@diat.ac.in

S. V. Dhavale
e-mail: sunitadhavale@diat.ac.in

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_66

1 Introduction

An individual's personality is made up of qualities, outlooks, and temperaments that influence his conduct and outlook on life. Automatic personality evaluation is required to improve human–computer interaction and to allow the user to self-introspect without the assistance of human specialists. For example, in the context of social engineering, the things and services recommended to a user could be those that have received positive feedback from others who share similar personality traits. Similarly, knowing personality features in forensics would allow security forces to predict suspects.

Our research is motivated by the findings of various researchers in the field of human affective behaviour [1–4]. We proposed to use a deep learning-based model to automatically recognise the apparent personality traits of individuals with only 15 s of interaction with the system. In the psychological literature, there are several definitions of personality modelling that exist, including 16 personality factors (16PF) [5], three-trait personality model (PEN) [6], Myers–Briggs type indicator (MBTI) [7] and Big-Five personality traits [8]. One of the most extensively used measures for automatic recognition of personality traits is the Big-Five personality traits model. It describes the personality of a person in five factors as openness, conscientiousness, extraversion, agreeableness and neuroticism (also acronym as OCEAN). A concise summary of the Big-Five factors are as follows:

- **Openness**: It is the extent to which a person exhibits self-awareness, artistic skills, imaginative, unconventional thinking, autonomous lifestyle, insightful, intelligent curiosity, etc.
- **Conscientiousness**: It is the extent to which an individual is planful, responsible, thorough, efficient, organised, dependable, reliable, etc.
- **Extraversion**: It is the extent to which a person is engaged with the outer world and possess the qualities of talkative, sociable, energetic, independent, active, assertive, zeal, etc.
- **Agreeableness**: It is the extent to which a person values kindness, honesty, social harmony, sympathetic towards teammates, trustworthy, appreciative, caring, compliant, etc.
- **Neuroticism**: It is the level of negative feelings and tendency to overreact in a person. The person appears as tense, touchy, hostile, unstable, worried, anxious, insecure, etc.

In this paper, we investigated the applicability of existing deep learning techniques for recognising Big-Five personality traits and proposed a fusion strategy (iSMART) that combines a convolution neural network (2D-CNN) and a long short-term memory with attention (LSTMwA) model for recognising apparent personality traits with bi-modal (audio and video) inputs. To summarise, the main contributions of this paper are as follows:

(a) Efficient and language-independent feature extraction methods are proposed for bi-modal inputs.

(b) Simulated an intelligent and lightweight CNN-LSTMwA model to effectively recognise Big-Five personality traits.
(c) Effectiveness of transfer learning approaches employing a pre-trained VGG network is investigated on the video dataset for personality recognition.
(d) The proposed system is compared to the state-of-the-art automatic personality recognition approaches.
(e) Identified the challenges and future directions for improving the automatic personality recognition model.

This paper is further organised as follows. We describe the related work towards bi-modal personality recognition in Sect. 2 followed by proposed methodology and novel architecture for training in Sect. 3. In Sect. 4, we describe the experimental setup and present the simulation results. Finally, we conclude the paper with challenges and recommendations for future work in the same context.

2 Related Work

Most of the previous research in the field of personality analysis has been observed using linguistic data [9]. However, the application of deep learning for the prediction of apparent personality traits has gained momentum for researchers since 2016 with the conduct of the ChaLearn LAP 2016 competition [10]. The methods adopted by top winning teams were a combination of deep learning models over handcrafted features in the bi-modal systems.

For apparent personality analysis, Zhang et al. presented the deep bi-modal regression (DBR) framework [11]. The framework came out on top of the ChaLearn Challenge 2016 podium. It is divided into three sections: visual modality, auditory modality and an ensemble component that combines the results of both modalities. By discarding the fully connected layers of the network, the modified architecture of the traditional CNN model is implemented as a descriptor aggregation network (DAN), after which the last convolution layer is both averaged and max pooled into 512-d feature vectors with standard L2 normalisation, followed by concatenation into a single combined 1024-d image representation. The log filter bank (logfbank) features are extracted directly from the original audios of each video for the audio modality. After that, a fc+sigmoid model was used to train the audio regressor. Thereafter, the regression loss was estimated using the L2 distance as the loss function. Finally, with late fusion, the two modalities are fused by averaging the visual and auditory model's five predicted values, yielding the final regression scores. The model took into account spatial elements, but it did not look at the videos' temporal patterns.

The second place finisher in the First Impression Challenge recommended the use of short videos to recognise Big-Five personality traits [12]. To predict the Big-Five personality traits, the authors proposed an LSTM-based model to extract the temporal patterns of images (from videos) and audio input. It uses audio and face images to describe two bi-modal deep CNNs. The first network uses 3D convolutional networks, whereas the second uses LSTM. Because of the 3D CNN, training the model on bi-modal input necessitates a lot of computing resources.

For perceived personality analysis from verbal and audio-visual data, the third winning team developed audio-visual deep residual networks (DRN) based on ResNet [13]. The auditory and visual subnetworks are identical to the deep residual network's first 17 layers. One convolution layer and eight residual blocks of two convolution layers make up each subnetwork. Batch normalisation, rectified linear units, max pooling (initial layer), and global average pooling follow each convolution layer (last layer). The auditory and visual streams' one-dimensional pooling kernels and strides are combined as audio-visual input to a fully connected layer. Because the fully connected layer is followed by hyperbolic tangent units, the audio-visual stream's outputs are scaled to [0, 1]. Since tanh is a logistic sigmoid function that varies from [−1,1], it would have resulted in dense activation and may have led to the vanishing gradient problem for such a personality-related dataset.

Many other researchers offered various approaches to recognise personality traits using the ChaLearn First Impression Dataset, following the trends of the ChaLearn Challenge 2016. For the examination of Big-Five personality traits, Yunan Li et al. presented a novel approach called classification–regression network (CR-Net) [14]. The proposed model uses ResNet34 as the backbone network first categorises personality traits before applying a regression, which results in personality trait prediction and interview recommendation. Because the analysis dataset contains continuous values in the range [0,1], and an individual's personality values are labelled for all five personality traits, using a classification method to such a purely regression situation would be inappropriate.

To address the concerns raised in the previous papers, we present iSMART, which combines a CNN-LSTM model (spatial and temporal fusion) with an attention mechanism to improve accuracy. Despite the limited computation and memory on a single GPU machine, we have developed a cost-effective and lightweight deep learning model that extracts relevant audio and video features from the ChaLearn dataset and then classifies them appropriately using depth-wise separable CNN (DSCNN) and LSTM with attention mechanisms (LSTMwA) in this paper.

3 Proposed Methodology

We present our methods for recognising personality traits using bi-modal inputs in this section. In this research, depth-wise separable CNNs are employed to build a lightweight model for personality trait recognition, partly inspired by the work of Chakraborty et al. [15]. To improve the development of relevant input requirements for this study, data pre-processing is performed in a series of steps for both audio and visual modalities. It is divided into two phases. The data samples are processed in the first phase to remove missing properties from raw data and eliminate individual differences, while the second phase generates audio and video subnetwork inputs.

3.1 Audio Data Pre-Processing

Using 'ffmpeg', we first extracted the audio from the video files. We utilised the PCM 32 bit floating-point little-endian format as an audio codec, with a single channel and a 44.1 kHz audio sample rate, which is commonly used for CD recording. For 128 mels filters with a sample rate of 22,050, the 'librosa' Python module is used to extract mel spectrogram features. Each audio frame is windowed (here, a window size of 2048 is used) and padded with zero to match the FFT window size. The estimated power spectrum is then transformed to decibel units before mel frequency cepstral coefficient (MFCC) with 24 coefficients is retrieved from the audio. MFCCs are also standardised sample by sample using zero mean and unit variance, as shown below:

$$\text{mfcc\_standardized} = \frac{\text{mfcc\_data} - \text{mean(mfcc\_data)}}{\text{Standard Deviation (mfcc}}\text{\_data)} \tag{1}$$

The length of the sample is then equalised by padding the standardised MFCCs with 0. Finally, the audio representation per sample is a tensor of shape (N, M, 1), with N denoting the number of coefficients and M being the number of audio frames. The experiment was carried out with N = 24 and M = 1320 in this publication.

3.2 Video Data Pre-processing

Using 'ffmpeg', count the number of frames in the video. The images are downsized to 140 × 248 to keep the aspect ratio of each frame. In addition, to reduce complexity, the frames are sub-sampled to six frames per video. Following that, a 128 × 128 random crop window is applied for training, while the centre crop has been used for validation and test data.

The random crop of images considered for training dataset is calculated as follows:

$$\text{MAX}_N(i) = \text{height}(i) - 128$$

$$\text{MAX}_M(i) = \text{width}(i) - 128$$

$$\text{rand}_N(i) = \text{randint}(0, \text{MAX}_N(i)$$

$$\text{rand}_M(i) = \text{randint}(0, \text{MAX}_M(i)$$

whereas the centre crop of images for validation and test sets is calculated as follows:

$$\text{index}_N(j) = (\text{height}(j) - 128)/2$$

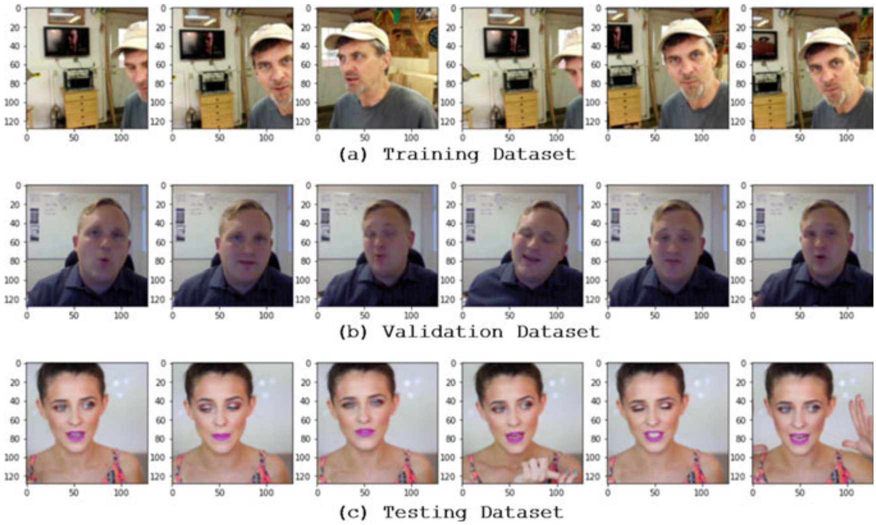


Fig. 1 Sample cropped images from dataset

$$\mathrm{index}_M(j) = (\mathrm{width}(j) - 128)/2$$

$$\mathrm{cropindex}_N(j) = \mathrm{index}_N(j) + 128$$

$$\mathrm{cropindex}_M(j) = \mathrm{index}_M(j) + 128$$

The sample cropped images from the ChaLearn LAP dataset are shown in Fig. 1. The video representation per sample in our experiment is modelled as a tensor with the shape (F, H, W, 3), where F denotes the number of frames and H and W denote the spatial dimensions. The training, validation and test sets are prepared for further processing based on the pre-processing approach. To iterate across the audio and visual data representations, a generator was constructed.

3.3 Architecture of 2D-CNN-LSTMwA Model

We present a lightweight model for personality trait recognition based on a depth-wise separable convolution neural network (DSCNN) and the long short-term memory with attention (LSTMwA) model in this research.

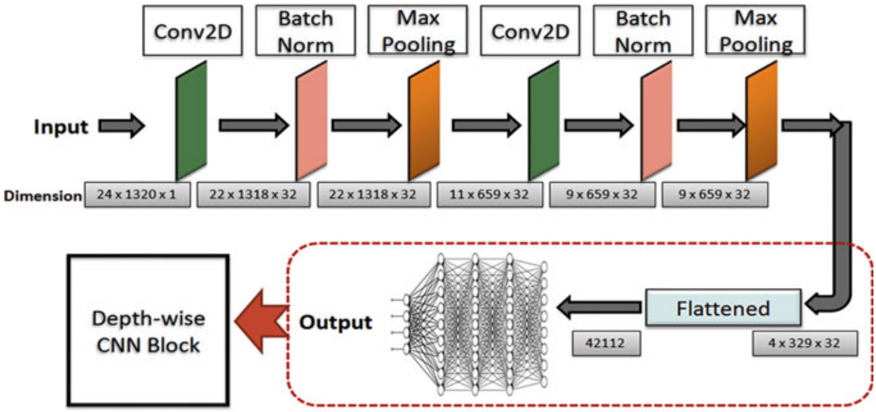


Fig. 2 Proposed Modification of Traditional Convolutional Neural Network

Depth-wise Separable Convolution Neural Network

For audio modal input, the proposed depth-wise 2D-CNN model is used. The suggested model's architecture and pipeline are depicted in Fig 2. To generate feature maps, the pre-processed audio data of size 24 × 1320 × 1 is first passed through a 2D convolution layer with 32 filters, kernel size (3 × 3), strides (1,1), padding as 'valid' and activation function as rectified linear unit (ReLU), then batch normalisation and a 2D max pooling layer of pool size (2X2) are applied. The resulting feature maps are then sent through a second set of similar 2D convolution layers, yielding a 4 × 328 × 32 feature map. Unlike traditional CNN, the output from the previous layer is now passed to a depth-wise separable convolution block that consists of five parallel paths, where multiple kernels are applied over the output from the previous layer, rather than flattening the feature dimension for fully connected layers.

Figure 3 depicts the design architecture of depth-wise separable CNN. Application of multiple different sized convolution filters to the same input to minimise overfitting, introduction of residual connections to reduce the effect of vanishing gradient, addition of batch normalisation layer to speed up the training and use of depth-wise separable convolution to reduce model parameters are some of the important design principles used in the architecture. There are five parallel channels that represent various convolution kernels that take input from the preceding layer's output with some pooling. Except for the first path, every path employs (1 × 1) convolution to assure dimension growth. In the third and fourth paths, (3 × 3) convolutions are used to extract multilevel feature representations. This method also aids in producing feature map dimensions that are similar to those provided in the input. In the fifth step, the max pooling layer is used to extract the input's low-level features. After that, the concatenated output is transferred to a convolution layer (1 × 1) for feature dimension reduction. Path one's result is residual learning, which is directly added to the concatenated output via ReLU activation.

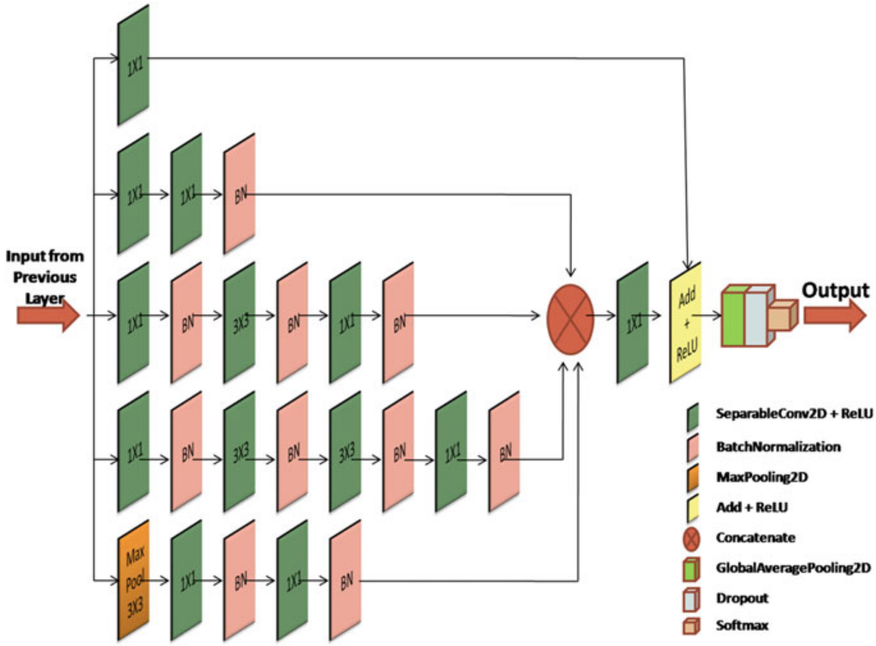


Fig. 3 Depth-wise separable convolutional neural network architecture [15]

Long Short-Term Memory with Attention (LSTMwA)

To capture the temporal patterns of each encoded frame, we propose using LSTM with an attention mechanism for video input. The suggested model's architecture and pipeline are depicted in Fig 4. Attention-based LSTM was proposed by Wang et al. [16] for sentiment classification, where attention is focused on different portions of the phrase according on the aspect under consideration. The efficacy of the attention mechanism for image categorisation was investigated by Lai et al. [14]. In the presence of extended sequences using time-variant context vectors, the encoder–decoder model has evolved with the usage of attention mechanism, which improves the performance of traditional models. The attention layer is used in this work to focus on the most relevant aspects for personality recognition. It aids in the selection of the most pertinent data from the source sequence. It is primarily made up of an encoder–decoder structure. The LSTM encoder is used to process the full input frame and encode it into a context vector, which forms the LSTM's final hidden state. The encoder's remaining intermediate stages are deleted, and the final state is considered the decoder's initial hidden state. The LSTM decoder generates predictions one by one in each frame. The attention mechanism gives a higher priority to a section of the image that contributes more to image analysis. To control overfitting, an LSTM

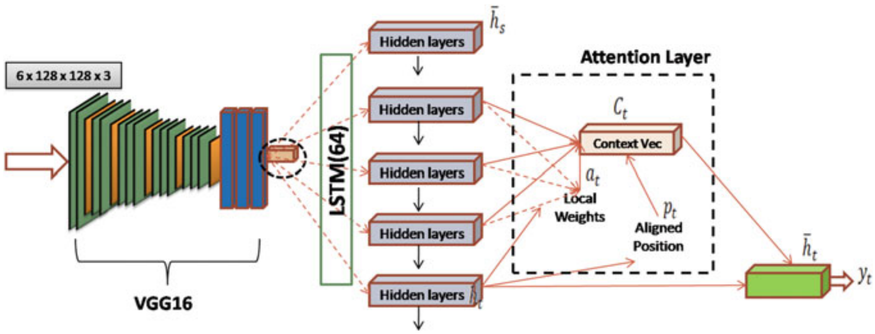


Fig. 4 Proposed attention-based LSTM for video analysis

with 64 modes is used with a dropout of 0.3. Following that, the input sequence is transmitted via an attention layer and then a fully connected layer to produce a 128-dimensional vector.

4 Experiments

4.1 Dataset

There have been very few datasets available in the open domain for research [17, 18] due to the merging of psychology and technology. In the realm of personality analysis, the ChaLearn First Impression Dataset [10] has dominated the researchers. In order to assess the proposed model's performance, we used the ChaLearn Looking at People First Impression Dataset to conduct personality recognition experiments. It is made up of $\approx$ 10000 HD video clips culled from 5563 YouTube videos in which someone speaks for about 15 s in front of the camera. Amazon Mechanical Turk (AMT) workers assisted the annotation of the dataset for Big-Five personality traits (i.e. OCEAN). Figure 5 depicts the probability density of the personality traits in the dataset.

The experiment's entire dataset is separated into three sections. The first portion is the training set, which contains 6000 video clips, the second part is the validation set, which contains 1998 video clips (2 of which were eliminated due to low quality), and the third part is the testing set, which contains 2000 video clips.

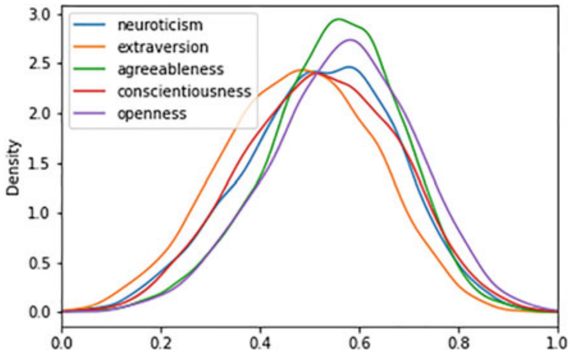


Fig. 5 Personality density graph of the dataset

4.2 Experiment Set-up

The experiment was carried out in two stages: development and evaluation. We trained the proposed model (both visual and auditory subnetworks) using the training set and subsequently confirmed its performance on the validation set during the development phase. We also suggested a video modal input transfer learning approach. We utilised VGG16 with weights learned on the 'ImageNet' dataset, which contains over 15 million high resolution and tagged images, because it had outperformed the other deep learning models in the ILSVRC contests. The entire architecture of the suggested model (iSMART) for personality trait recognition is shown in Fig 6.

The time-distributed video frames of size $6 \times 128 \times 128 \times 3$ (RGB image) are passed to a pre-trained model (VGG16 with 'ImageNet') to get the encoded frame of 512-dimensional vector for encoder LSTM, and the pre-processed audio files of size $24 \times 1320 \times 1$ are passed to 2D-CNN followed by depth-wise separable CNN to get 128-dimensional output. To generate the intended output as a score for the Big-Five

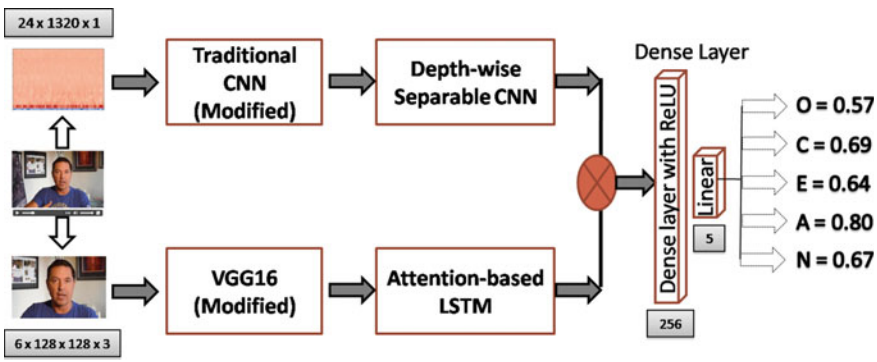


Fig. 6 Proposed model (iSMART) for recognition of personality traits

personality traits, we concatenated the final hidden representations of the audio and visual subnetworks and added fully connected layers (256 units, ReLU) and another dense layer (5 units, linear) at the end.

The experiment is performed with a mini-batch size of 32 and a 0.001 learning rate. The development phase was run for 20 epochs on a single GPU system with an early stop mechanism to prevent overfitting. The proposed model takes almost 96 h to train and validate. The ideal models from the development phase are then employed in the evaluation phase to predict personality scores using the testing set data.

4.3 Experiment Results

We employed a test set for the evaluation of the suggested model (2000 video clips with corresponding traits values). 1-mean absolute error is used as the evaluation metric (1-MAE). The proposed model achieves values around 0.91 because it is an un-normalised metric and the target variables have a Gaussian distribution.

$$1 - \mathrm{MAE} = 1 - \left(\frac{1}{n}\right)\sum_{i=1}^{n} |y_i - x_i| \tag{2}$$

where y_i is the predicted value and x_i is the true value.

4.3.1 Comparison of Test Accuracy

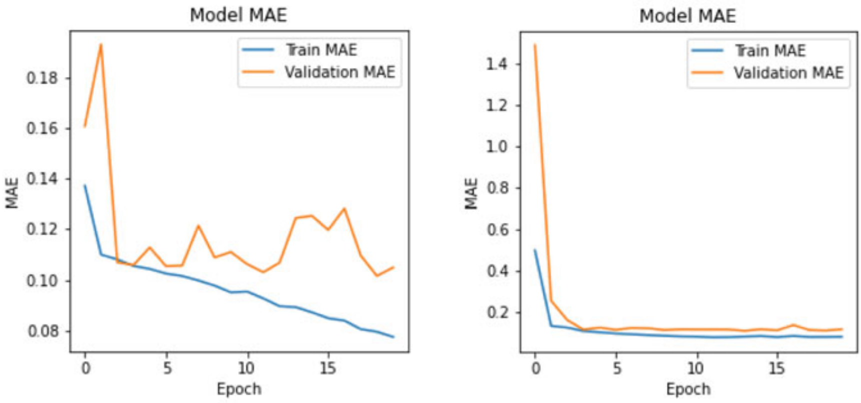
Table 1 shows a comparison matrix of state-of-the-art algorithms based on deep learning models with bi-modal inputs on the first impression, ChaLearn Dataset [10]. Averaging the 1-MAE value of each personality trait yielded the average accuracy.

The outcome clearly demonstrates that the suggested model outperforms existing state-of-the-art approaches. Figure 7 shows line plots of mean absolute error loss

Table 1 Comparison of results for recognition of personality traits

Methods	Avg Acc.	Extra.	Agree.	Consc.	Neuro.	Open.
iSMART (ours)	0.9166	0.9215	0.9170	0.9150	0.9125	0.9170
NJU-LAMBDA[11]	0.9129	0.9133	0.9126	0.9166	0.9100	0.9123
Evolgen[12]	0.9120	0.9150	0.9119	0.9119	0.9099	0.9117
BU-NKU[19]	0.9094	0.9161	0.9070	0.9133	0.9021	0.9084
DBR-LSTM[20]	0.8963	0.8977	0.8977	0.8941	0.9033	0.8888

[a] Average accuracy is taken from 1-MAE of each personality traits



(a) CNN-LSTMwA (b) CNN-BiLSTM

Fig. 7 Line plots of MAE loss over training epochs of the model using CNN-LSTMwA and CNN-BiLSTM model

Table 2 Analysis of trainable parameters for CNN + LSTM model

Methods	Total parameters	Trainable parameters	Non-trainable	Evaluation (MAE)
Depth-wise CNN + LSTMwA	961,454	958,318	3136	0.0894
CNN + BiLSTM	5,746,405	5,746,277	128	0.1061
2D-CNN + LSTM	5,582,309	5,582,181	128	0.1045

over training epochs of the model. It demonstrates that the training and validation losses are approaching each other and are not causing overfitting or underfitting.

4.3.2 Comparison of Trainable Parameters in Various Models

The experiment was conducted with many variants of the proposed model. The comparison of total trainable parameters and accuracy obtained is shown in Table 2. The number of trainable parameters has been remarkably reduced with the use of proposed depth-wise separable CNN model.

5 Conclusion and Future Work

Considering the further perspectives for recognition of personality traits and its effect on human–computer interaction, the requirement for profound research is felt fundamental. As we have introduced in this paper, deep learning techniques are exceptionally productive in the recognition of personality traits even using a brief video clip ($\approx$ 15 s). We have acquired a superior outcome that is similar to the top rivals in the ChaLearn Challenge utilising DSCNN-LSTMwA model. It likewise creates the impression that the proposed technique utilising attention mechanism has significantly high potential to additionally improve the test precision, whenever executed for any extra modular data sources like transcriptions of sounds, gender and age of the speaker and so on. Bidirectional LSTM may likewise be applied to upgrade the precision of the model. The effect of sentiments/facial emotions for the recognition of personality traits will be investigated in future.

Acknowledgements We would like to acknowledge the valuable feedback and suggestion of Dr. Suresh Kumar, Scientist at Defence Institute of Psychological Research, New Delhi. The authors also recognise the support of fellow research scholars and staff of Defence Institute of Advanced Technology, DRDO Laboratory, Ministry of Defence, India.

References

1. Xia F, Asabere NY, Liu H, Chen Z, Wang W (2017) Socially aware conference participant recommendation with personality traits. IEEE Syst J 11(4):2255–2266. https://doi.org/10.1109/JSYST.2014.2342375
2. Willis J, Todorov A (2006) First impressions: making up your mind after a 100-ms exposure to a face. Psychol Sci 17(7):592–598. https://doi.org/10.1111/j.1467-9280.2006.01750.x. PMID: 16866745
3. Aslan S, Gudukbay U (2019) Multimodal video-based apparent personality recognition using long short-term memory and convolutional neural networks
4. Ali I (2019) Personality traits, individual innovativeness and satisfaction with life. J Innov Knowl 4(1):38–46. https://doi.org/10.1016/j.jik.2017.11.002
5. Cattell HEP, Mead AD (2008) The sixteen personality factor questionnaire (16pf). In: Boyle GJ, Matthews G, Saklofske DH (eds) The SAGE handbook of personality theory and assessment, vol 2, pp 135–159 (2008). https://doi.org/10.4135/9781849200479.n7, https://psycnet.apa.org/record/2008-14475-007
6. William Revelle DMC (2015) A model for personality at three levels. J Res Personal 56:70–81. https://doi.org/10.1016/j.jrp.2014.12.006, https://www.sciencedirect.com/science/article/pii/S0092656615000318
7. Michael J (2003) Using the Myers-Briggs type indicator as a tool for leadership development? Apply with caution. J Leader Organ Stud 10(1):68–81. https://doi.org/10.1177/107179190301000106
8. Digman J (1990) Personality structure: emergence of the five-factor model. Ann Rev Psychol 41:417–440. https://doi.org/10.1146/annurev.ps.41.020190.002221
9. Mairesse F, Walker M (2007) PERSONAGE: personality generation for dialogue. In: Proceedings of the 45th annual meeting of the association of computational linguistics. Association for Computational Linguistics, Prague, Czech Republic, pp 496–503. https://aclanthology.org/P07-1063
10. Ponce-López C (2016) ChaLearn LAP 2016: first round challenge on first impressions–dataset and results. Springer International Publishing. https://doi.org/10.1007/978-3-319-49409-8_32
11. Wei XS, Zhang CL, Zhang H, Wu J (2017) Deep bimodal regression of apparent personality traits from short video sequences. IEEE Trans Affective Comput 9(3):303–315. https://doi.org/10.1109/TAFFC.2017.2762299

12. Subramaniam A, Patel V, Mishra A, Balasubramanian P, Mittal A (2016) Bi-modal first impressions recognition using temporally ordered deep audio and stochastic visual features, pp 337–348
13. Güçlütürk YGX, Escalante HJ, Guyon I, Escalera S, van Gerven MAJ, van Lier R (2018) Multimodal first impression analysis with deep residual networks. IEEE Trans Affect Comput 9(3):316–329. https://doi.org/10.1109/TAFFC.2017.2751469
14. Li Y, Wan J, Miao Q, Escalera S, Fang H, Chen H, Ali M, Guo G (2020) Cr-net: a deep classification-regression network for multimodal apparent personality analysis. Int J Comput Vis 128. https://doi.org/10.1007/s11263-020-01309-y
15. Dhavale SV, Ingole J, Chakraborty M (2021) Corona-Nidaan: lightweight deep convolutional neural network for chest X-Ray based COVID-19 infection detection. Appl Intell 51:3026–3043. https://doi.org/10.1007/s10489-020-01978-9
16. Wang Y, Huang M, Zhu X, Zhao L (2016) Attention-based lstm for aspect-level sentiment classification, pp 606–615. https://doi.org/10.18653/v1/D16-1058
17. Mehta Y, Majumder N, Gelbukh A, Cambria E (2020) Recent trends in deep learning based personality detection. Artif Intell Rev 53. https://doi.org/10.1007/s10462-019-09770-z
18. Giritlioglu D, Mandıra B, Yilmaz S, Ertenli U, Akgür B, Kurt AG, Mutlu E, Gürel EC, Dibeklioglu H (2020) Multimodal analysis of personality traits on videos of self-presentation and induced behavior. J Multimodal User Interf 15. https://doi.org/10.1007/s12193-020-00347-7
19. Gürpinar F, Kaya H, Salah AA (2016) Multimodal fusion of audio, scene, and face features for first impression estimation. In: 23rd international conference on pattern recognition (ICPR), pp 43–48 (2016). https://doi.org/10.1109/ICPR.2016.7899605
20. Stanford KY, Mall S, Stanford NG (2017) Prediction of personality first impressions with deep bimodal LSTM

Correction to: Occlusion Problem in 3D Object Detection: A Review

Apurva Kandelkar, Isha Batra, Shabnam Sharma, and Arun Malik

Correction to: Chapter "Occlusion Problem in 3D Object Detection: A Review" in: D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473, https://doi.org/10.1007/978-981-19-2821-5_26

In the original version of the chapter, the following correction has been incorporated:

In Chapter 26, the affiliation of the author "Isha Batra" has been changed from "CMR University, Bengaluru, Karnataka, India" to "Lovely Professional University, Phagwara, Punjab 144001, India" and the affiliation of the author "Shabnam Sharma" has been changed from "Lovely Professional University, Phagwara, Punjab 144001, India" to "CMR University, Bengaluru, Karnataka India".

The Correction chapter and the book have been updated with the change.

The updated version of this chapter can be found at
https://doi.org/10.1007/978-981-19-2821-5_26

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5_67

Author Index

D. Gupta et al. (eds.), *International Conference on Innovative Computing and Communications*, Lecture Notes in Networks and Systems 473,
https://doi.org/10.1007/978-981-19-2821-5