Siba K. Udgata
Srinivas Sethi
Xiao-Zhi Gao   *Editors*

Intelligent Systems

Proceedings of ICMIB 2021

Springer

Lecture Notes in Networks and Systems

Volume 431

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

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

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

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

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

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

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

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

Siba K. Udgata · Srinivas Sethi · Xiao-Zhi Gao
Editors

Intelligent Systems

Proceedings of ICMIB 2021

Springer

Editors
Siba K. Udgata
School of Computer and Information
Science
University of Hyderabad
Hyderabad, Telangana, India

Srinivas Sethi
Department of Computer Science
and Engineering
Indira Gandhi Institute of Technology
Sarang, India

Xiao-Zhi Gao
Faculty of Science and Forestry
School of Computing
University of Eastern Finland
Kuopio, Finland

Committee Members

Patron

Prof. Satyabrata Mohanto, Director, IGIT, Sarang

General Chair

Prof. Lalit Mohan Patnaik, National Institute of Advanced Studies and Indian Institute of Science, Bangalore

Program Chair

Prof. Siba K. Udgata, University of Hyderabad, India

Program Co-chairs

Prof. Srinivas Sethi, IGIT, Sarang
Prof. Xiao-Zhi Gao, University of Eastern Finland, Finland

Organizing Chairs

Prof. S. N. Mishra, IGIT, Sarang
Prof. Srinivas Sethi, IGIT, Sarang

Convenor

Prof. Srinivas Sethi, IGIT, Sarang

Publicity Chairs

Prof. B. P. Panigrahi, IGIT, Sarang
Prof. Sasmita Mishra, IGIT, Sarang
Prof. Ashima Rout, IGIT, Sarang
Prof. S. K. Tripathy, IGIT, Sarang
Prof. Subhrashu Das, GCE, Keunjhar

Hospitality Chairs

Prof. Sanjaya Patra, IGIT, Sarang
Prof. Biswanath Sethi, IGIT, Sarang

Finance Chair

Prof. Sanjaya Patra, IGIT, Sarang

Technical Program Committee

Manas Ranjan Patra, Berhampur University
Siba K. Udgata, University of Hyderabad
Soodkhet Projpapi, Suranaree University of Technology, Thailand
Mufti Muhamad, University of Trent, UK
Md. Atiqur Rahman Ahad, University of Osaka, Japan
Shaik Shakeel Ahamad, Majmaah University, Saudi Arabia
O. B. V. Ramanaiah, JNTU, Hyderabad
Jay Bagga, Ball State University, USA
Lalit Garg, University of Malta, Malta
Xiao-Zhi Gao, University of Eastern Finland, Finland
G. Suvarna Kumar, MVGRCE, Vijayanagaram
G. Sandhya, MVGRCE, Vijayanagaram
R. Hemalatha, University College of Engineering, Osmania University

Amit Kumar Mishra, University of Capetown, South Africa
R. Thangarajan, Kongu Engineering College, Tamil Nadu
Birendra Biswal, Gayatri Vidya Parishad College of Engineering, Vishakhapatnam
Somanath Tripathy, IIT Patna
A. K. Turuk, NIT Rourkela
B. D. Sahoo, NIT Rourkela
D. P. Mohapatra, NIT Rourkela
P. M. Khilar, NIT Rourkela
P. G. Sapna, CIT, Coimbatore
R. K. Dash, NIC, Mizoram
B. K. Tripathy, VIT, Vellore
Moumita Patra, IIT Guwahati
S. N. Das, GIET University, Gunupur, Odisha
Ram Kumar Dhurkari, IIM, Sirmaur, Himachal Pradesh
Chitta Ranja Hota, BITS Pilani, Hyderabad
A. Kavitha, JNTU, Hyderabad
Subasish Mohapatra, CET, Bhubaneswar
Sanjaya Kumar Panda, NIT Warangal
S. Mini, NIT Goa
S. Nagender Kumar, University of Hyderabad
Lalitha Krishna, Kongu Engineering College, Tamil Nadu
C. Poongodi, Kongu Engineering College, Tamil Nadu
Sumanth Yenduri, Kennesaw University, USA
K. Srujan Raju, CMR Technical Campus, Hyderabad
R. Hemalatha, University College of Engineering, Osmania University, Hyderabad
P. Sakthivel, Anna University
Pavan Kumar Mishra, NIT Raipur
Tapan Kumar Gandhi, Department of Electrical Engineering, IIT Delhi
Annappa B., NIT Surathkal
Prafulla Kumar Behera, Utkal University, Bhubaneswar, Odisha
Nekuri Naveen, School of Computer and Information Sciences, University of Hyderabad
Ch. Venkaiah, School of Computer and Information Sciences, University of Hyderabad
Rajendra Lal, School of Computer and Information Sciences, University of Hyderabad
Dillip Singh Sisodia, Department of Computer Science and Engineering, NIT Raipur
Pradeep Singh, Department of Computer Science and Engineering, NIT Raipur
Satessh Kumar Pradhan, Utkal University, Bhubaneswar, India
Sumagna Patnaik, JB Institute of Engineering and Technology, Hyderabad
Ajit K. Sahoo, University of Hyderabad
Atluri Rahul, Neurolus Systems, Hyderabad
Samrat L. Sabat, Center for Advanced Studies in Electronic Science and Technology (CASEST), University of Hyderabad
Nihar Satapathy, Sambalpur University

Rabindra Behera, IGIT, Sarang
L. N. Tripathy, CET, Bhubaneswar
B. B. Choudhary, IGIT, Sarang
Dhiren Behera, IGIT, Sarang
R. N. Sethi, IGIT, Sarang
Anand Gupta, IGIT, Sarang
Ayaskanta Swain, NIT Rourkela
Gayadhr Panda, NIT Meghalaya
S. K. Tripathy, IGIT, Sarang
B. B. Panda, IGIT, Sarang
Md. N. Khan, IGIT, Sarang
Anukul Padhi, IGIT, Sarang
Debakanta Tripathy, IGIT, Sarang
S. K. Maity, IGIT, Sarang
Devi Acharya, VIT, Vellore
Sanjaya Kumar, PRSU, Raipur, India
V. Patle, PRSU, Raipur, India
Saurov Bhoi, PMEC Berhampur
Kalyan Kumar Jena, PMEC Berhampur
Alok Ranjan Prusty, Skill Development and Entrepreneurship, India
Babita Majhi, GGU, Chattisgarh
Subhrashu Das, GCE, Keunjhar
Puspalata Pujahari, GGU, Chattisgarh
Tirimula Rao, JNTU Kakinada
K. Shirsagar Sahoo, SRM University, Amaravati
Maheswar Behera, IGIT, Sarang
Niranjan Panigrahy, PMEC, Berhampur
Trilochon Rout, PMEC, Berhampur
P. K. Panigrahy, GIET, India
Mihir Kumar Sutar, UCE, Burla
Sukant Kishro Bisoy, C. V. Raman Global University, Bhubaneswar

List of Reviewers

Ajit Kumar Sahoo
Anand Agrawal
Anitha A.
Ansuman Padhi
Anupama Sahu
Arun Avinash Chauhan
Arunima Pattanayak
Ashima Rout
Ashutosh Bhoi
Babita Majhi
Bibhudatta Sahoo
Binayak Das
Biswanath Sethi
Bonomali Khuntia
Byomakesh Mahapatra
Chinmayee Rout
Chiranji Lal Chowdhary
Chittaranjan Pradhan
Debabrata Dansana
Debasis Mohapatra
Debendra Muduli
Deva Priya M.
Dilip Senapati
Dillip Ghose
Diptendu S. Roy
Dr. Pramod Kumar Parida
Alok Ranjan Prusty
Alok Ranjan Tripathy
Bibudhendu Pati
Chhabi Rani Panigrahi
Chinmaya Sahu

Sanjaya Kumar Panda
Sukant Kihsoro Bisoy
Suvendra Kumar Jayasingh
Dr. Devesh Bandil
Gopal Behera
J. Chandrakanta Badajena
Jibendu Mantri
Jitendra Kumar Rout
Jui Pattnayak
K. Hemant Kumar Reddy
Kaibalya Prasad Panda
Kali Rath
Kallam Suresh
Kalyan Kumar Jena
Kauser Ahmed
Kavitha A.
Kavitha Athota
Krushna Chandra Sethi
Kshira Sahoo
Lalitha Ananth
M. Mishra
M. Saifullah
Madhusmita Panda
Maheswar Behera
Manmath Kumar Bhuyan
Manoj Kumar Muni
Manoj Kumar Patra
Mihir Kumar Sutar
Minakhi Rout
Mishra M.
Mrs. Bichitra Mandal
Niranjan Panigrahi
Nitin Bommi
Pavan Mishra
Prahallad Kumar Sahu
Prajna Paramita Nanda
Prasant Ranjan Dhal
Pritee Parwekar
Prof. Punyaban Patel
Prof. Sunil Kumar Tripathy
Purna Chandra Sethi
Pushkar Kishore
Rajendra Nayak
Rajiv Senapati
Rakesh Swain

Ramesh K. Sahoo
Ravi Sankar Barpanda
Rohit Kumar Bondugula
Sagarika Mohanty
Salman Abdul Moiz
Sambit Kumar Mishra
Sampa Sahoo
Sangita Pal
Sanjeeb Mohanty
Sanjeet Nayak
Santanu Kumar Dash
Santos Kumar Baliarsingh
Saraju Prasad Padhy
Saroj Kumar Panigrahy
Soubhagya Barpanda
Sourav Bhoi
Srichandan Sobhanayak
Sruthi P.
Subasish Mohapatra
Subhasish Pani
Subhransu Dash
Suman Paul
Sumit Kar
Suraj Sharma
Suryalok Dash
Susil Kumar Mohanty
Swarup Roy
Tapan Kumar Mahanta
Tirimula Rao Benala
Umakanta Samantsinghar

Preface

This Springer LNNS volume contains the papers presented at the 2nd International Conference on Machine Learning, Internet of Things and Big Data (ICMIB-2021) held during December 18 to 20, 2021, organized by Indira Gandhi Institute of Technology (IGIT), Sarang, Odisha, India. The pandemic threw a lot of challenges at us, and no words of appreciation are enough for the organizing committee who could still pull it off successfully.

The conference draws some excellent technical keynote talks and papers. Two tutorial talks by Prof. Bibhudatta Sahoo, NIT Rourkela, and Prof. Nagender Kumar Suryadevara, University of Hyderabad, India, were conducted on December 18, 2021. The overwhelming response for the tutorial talks is worth mentioning. Apart from the tutorial sessions, five keynote talks by Prof. Lalit Mohan Patnaik (National Institute of Advanced Studies and Indian Institute of Science, Bangalore, India), Prof. Mufti Mahmud (Nottingham Trent University, UK), Prof. Amit Mishra (University of Cape Town, South Africa), Dr. Tinku Acharya (IEEE Fellow) from Videonetics, Dr. Sahasranamam Subramanian (Semic RF, Germany, and GAPVC, Global Asia Partners) are delivered. We are grateful to all the speakers for accepting our invitation and sparing their time to deliver the talks.

For the conference, in spite of the adverse situation and lockdown throughout the world, we received 167 full paper submissions and we accepted only 59 papers. The contributing authors are from different parts of the globe that includes USA, Iceland, Saudi Arabia, Malaysia, Algeria, Pakistan, Papua New Guinea, and India. All the papers are reviewed by at least three independent reviewers and in some cases by as many as five reviewers. All the papers are also checked for plagiarism and similarity score. It was really a tough job for us to select the best papers out of so many good papers for presentation in the conference. We had to do this unpleasant task, keeping Springer guidelines and approval conditions in view. We take this opportunity to thank all the authors for their excellent work and also the reviewers who have done an excellent job.

On behalf of the technical committee, we are indebted to Prof. L. M. Patnaik, General Chair of the Conference, for his timely and valuable advice. We cannot

imagine the conference without his active support at all the crossroads of decision-making process. The management of the host institute, particularly Director Prof. Satyabrata Mohanta; HOD, CSEA, Prof. (Mrs.) Sasmita Mishra; Organizing Chair and Convenor Prof. Srinivas Sethi; and Organizing Chair Prof. S. N. Mishra, has extended all possible support for the smooth conduct of the conference. Our sincere thanks to all of them.

We would also like to place on record our thanks to all the keynote speakers, tutorial speakers, reviewers, session chairs, authors, technical program committee members, various chairs to handle finance, accommodation, and publicity and above all to several volunteers.

Our sincere thanks to all press, print, and electronic media for their excellent coverage of this conference.

We are also thankful to Springer Nature publication house for agreeing to publish the accepted papers in their Lecture Notes in Networks and Systems (LNNS) series.

Please take care of yourself and your loved ones and stay safe.

Hyderabad, India	Dr. Siba K. Udgata
Sarang, India	Dr. Srinivas Sethi
Kuopio, Finland	Dr. Xiao-Zhi Gao
December 2021	

Contents

Editors and Contributors

About the Editors

Prof. Siba K. Udgata is Professor in Computer and Information Sciences at University of Hyderabad, India. He has a Ph.D. in Computer Science in the area of mobile computing and wireless communications and also worked as United Nations Fellow and worked in the UNU/IIST, Macau. His research focus is on wireless communication, mobile computing, intelligent sensors, sensor network algorithms, Internet of Things, and applications. He was Volume Editor for several Springer LNAI, AISC International Conference proceedings, and also Associate Editor and Editorial Board Member of IOS Press *KES Journal* and Elsevier *AKCE International Journal of Graphs and Combinatorics*. Prof. Udgata has published more than 100 research papers in reputed international journals and conference proceedings. He has worked as Principal Investigator in many Government of India funded research projects mainly for the development of wireless sensor network applications, network security-related applications, and application of swarm intelligence techniques in the cognitive radio network domain.

Dr. Srinivas Sethi is Professor and has been actively involved in teaching and research in Computer Science since 1997. He did his Ph.D., in the area of Mobile Ad hoc Network, and is also continuing his work in wireless communication, sensor network, cognitive radio network, IOT, BCI, and cloud computing. He was Volume Editor for Springer LNNS, International Conference proceedings, Board for different journal and Program Committee Member for different international conferences/workshop. Now he is working as Faculty in the Department of Computer Science Engineering and Application at Indira Gandhi Institute of Technology Sarang, India, and has published more than 80 research papers in International journals and conference proceedings. He completed 7 research projects funded by different Government of India funding agencies such as DST, AICTE, NPIU, and DRDO.

Prof. Xiao-Zhi Gao is Full Professor at School of Computing, University of Eastern Finland, Finland. He heads the nature-inspired machine learning research group at UEF. His main research interests are bio-inspired computing, nature-inspired computing, machine learning, soft computing, data mining, and communication networks. He has published more than 400 papers in different international journals and conferences of repute. He has been associated with many international conference as Technical Program Co-chair, member and editorial board member also.

Contributors

N. Abdullah Faculty of Applied and Human Sciences, Institute of Engineering Mathematics, Universiti Malaysia Perlis, Arau, Perlis, Malaysia

Sasmita Acharya Department of Computer Application, VSSUT, Burla, India

Nibedita Adhikari Centre for Advanced Post Graduate Studies, Biju Patnaik University of Technology, Rourkela, Odisha, India

Aditya Agrawalla Department of Computer Science Engineering and Applications, Indira Gandhi Institute of Technology, Sarang, Dhenkanal, Odisha, India

Mahmoud Al Ahmad School of Computer Engineering, KIIT Deemed to be University, Bhubaneswar, India

F. Ajesh Sree Buddha College of Engineering, Alappuzha, Kerala, India

Sahoo Ajit Kumar School of Computer and Information Sciences, University of Hyderabad, Hyderabad, India

K. Akhil School of Computer and Information Sciences, University of Hyderabad, Hyderabad, India

Saeed Al-Amodi School of Computer Engineering, KIIT Deemed to be University, Bhubaneswar, India

Jawaz Alam Department of Mechanical Engineering, VSSUT, Burla, Odisha, India

S. D. Ambadkar G H Raisoni University, Amravati, India

D. Anil Kumar GIET University, Gunupur, Odisha, India

K. Anwar Faculty of Applied and Human Sciences, Institute of Engineering Mathematics, Universiti Malaysia Perlis, Arau, Perlis, Malaysia

Hemant Kumar Apat National Institute of Technology, Rourkela, Rourkela, India

A. Arun Prakash Department of Electronics and Communication Engineering, Kongu Engineering College, Erode, India

J. S. Arunalatha University Visvesvaraya College of Engineering, Bangalore University, Bengaluru, India;
Bangalore University, Bengaluru, India

D. S. Aswar Sinhgad College of Engineering, Pune, India

Jenamani Chandrakanta Badajena Utkal University, Bhubaneswar, Odisha, India;
Indira Gandhi Institute of Technology, Sarang, India

Akash Barik Department of Computer Science, New Horizon College of Engineering, Bengaluru, India

Lalbihari Barik Department of Information Systems, Faculty of Computing and Information Technology, King Abdulaziz University, Rabigh, Kingdom of Saudi Arabia

Rabindra K. Barik School of Computer Applications, KIIT Deemed to be University, Bhubaneswar, India

Gopal Behera Department of Computer Science and Engineering, Government College of Engineering Kalahandi, Bhawanipatna, Odisha, India

Sibananda Behera Department of MCA, Trident Academy of Creative Technology, Bhubaneswar, India

Dinabandhu Bhandari Department of Computer Science and Engineering, Heritage Institute of Technology, Kolkata, India

Ashok Kumar Bhoi Department of Computer Science and Engineering, Government College of Engineering Kalahandi, Bhawanipatna, Odisha, India

Ashutosh Bhoi International Institute of Information Technology, Bhubaneswar, India

Raju Bhoyar Department of Electrical Engineering, G H Raisoni University, Amravati, India

M. Bhuyan Department of Physics, Faculty of Science, Center for Theoretical and Computational Physics, University of Malaya, Kuala Lumpur, Malaysia;
Institute of Research and Development, Duy Tan University, Da Nang, Vietnam

Shubhanshu Biswal Odisha University of Technology and Research, Bhubaneswar, Odisha, India

Debanshu Biswas School of Computer Engineering, Kalinga Institute of Industrial Technology Deemed to be University, Bhubaneswar, India

Nitin Sai Bommi University of Hyderabad, Hyderabad, India

Sarath Kumar Bommi Indian Institute of Technology Jodhpur, Jodhpur, India

Rohit Kumar Bondugula School of Computer and Information Sciences, University of Hyderabad, Hyderabad, India

Arun Avinash Chauhan School of Computer and Information Sciences, University of Hyderabad, Hyderabad, India

Sanjoy Choudhury National Institute of Technology, Meghalaya, Shillong, India; S. N. Bose National Centre for Basic Sciences, Kolkata, India

A. Christy Jeba Malar Department of Information Technology, Sri Krishna College of Technology, Kovaipudur, Coimbatore, Tamilnadu, India

Binayak Das TCS Bhubaneswar, Bhubaneswar, Odisha, India

Jayanta K. Das Department of Pediatrics, Johns Hopkins University School of Medicine, Baltimore, MD, USA

Madhab Chandra Das School of Electrical Engineering, KIIT University, Bhubaneswar, Odisha, India

Madhusmita Das Institute of Technical Education and Research, Siksha 'O' Anusandhan, Deemed to be University, Bhubaneswar, India

Subhalaxmi Das Deparment of Computer Science and Applications, Utkal University, Bhubaneswar, India

Suchismita Das CSE Department, International Institute of Information Technology, Bhubaneswar, Odisha, India;
CSE Department, KIIT Deemed to be University, Bhubaneswar, Odisha, India

Amrit Dash Indira Gandhi Institute of Technology, Sarang, India

Rasmita Dash Institute of Technical Education and Research, Siksha 'O' Anusandhan, Deemed to be University, Bhubaneswar, India

Sanjit Kumar Dash Odisha University of Technology and Research, Bhubaneswar, Odisha, India

Subhransu Sekhar Dash Department of Electrical Engineering, Government College of Engineering, Keonjhar, Odisha, India

Rajat U. Davangeri Department of CSE, Global Academy of Technology, Bengaluru, Karnataka, India

Pidiyar Dharani Deepika Department of Electrical and Electronics Engineering, Sreenidhi Institute of Science and Technology, Hyderabad, India

M. Deva Priya Department of Computer Science and Engineering, Sri Eshwar College of Engineering, Coimbatore, Tamilnadu, India

Seema Dewangan Guru Ghasidas Vishwavidyalaya (A Central University), Bilaspur, Chhattisgarh, India

Sourav Dey Department of CSE, Global Academy of Technology, Bengaluru, Karnataka, India

Prasant Ranjan Dhal Department of Mechanical Engineering, Indira Gandhi Institute of Technology, Sarang, Odisha, India

P. Divya Department of IT, Kongu Engineering College, Perundurai, India

Sumit Dutta Network Reconstruction & Analysis (NetRA) Lab, Department of Computer Applications, Sikkim University, Gangtok, Sikkim, India

Dillip Kumar Ghose Department of Civil Engineering, National Institute of Technology Silchar, Assam, India

Sourav Ghose Department of Civil Engineering, Veer Surendra Sai University of Technology, Burla, Odisha, India

Priyom Goswami Subir Chowdhury School of Quality and Reliability, IIT Kharagpur, Kharagpur, West Bengal, India

Dr. Anand Gupta Department of Mechanical Engineering, Indira Gandhi Institute of Technology, Sarang, India

K. Hemant Kumar Reddy NITTE, Mangaluru, India

D. Ishwarya Department of Electronics and Communication Engineering, Kongu Engineering College, Erode, India

Nishu Jain School of Physics and Materials Science, Thapar Institute of Engineering and Technology, Patiala, Punjab, India

Suvendra Kumar Jayasingh Department of Computer Application, Maharaja Sriram Chandra Bhanja Deo University, Baripada, Odisha, India

A. Jeevanantham Department of IT, Kongu Engineering College, Perundurai, India

Kannan Kaliappan Electrical and Electronics Engineering, Sreenidhi Institute of Science and Technology, Hyderabad, India

M. Kandpal KIIT Deemed to be University, Bhubaneswar, India

Khushboo Kanodia Nokia Networks, Bangalore, India

Sanjeeb Kumar Kar Department of Electrical Engineering ITER, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha, India

B. H. Karthik Pai NITTE, Mangaluru, India

P. Kowshika Department of IT, Kongu Engineering College, Perundurai, India

Raj Kumar School of Physics and Materials Science, Thapar Institute of Engineering and Technology, Patiala, Punjab, India

Ranjith Kumar Electrical and Electronics Engineering, Sreenidhi Institute of Science and Technology, Hyderabad, India

Saroj Kumar Robotics Laboratory, Department of Mechanical Engineering, National Institute of Technology, Rourkela, Odisha, India

K. Lalitha Department of IT, Kongu Engineering College, Perundurai, India

Shaheen Layaq Department of Computer Science, Kakatiya University, Warangal, Telangana, India

Joshua T. Majekodunmi Faculty of Applied and Human Sciences, Institute of Engineering Mathematics, Universiti Malaysia Perlis, Arau, Perlis, Malaysia

Arpita Majhi Department of Computer Science and Application, Utkal University, Bhubaneswar, Odisha, India

Ravikanth Mallajoshula Department of Electrical and Electronics Engineering, Sreenidhi Institute of Science and Technology, Hyderabad, India

Mallikarjun University Visvesvaraya College of Engineering, Bangalore University, Bengaluru, India;
Bangalore University, Bengaluru, India

B. Manjula Department of Computer Science, Kakatiya University, Warangal, Telangana, India

Jibendu Kumar Mantri Department of Computer Application, Maharaja Sriram Chandra Bhanja Deo University, Baripada, Odisha, India

Bibhu Prasad Mishra Department of Civil Engineering, Odisha University of Technology and Research, Bhubaneswar, Odisha, India

Jibitesh Mishra Odisha University of Technology and Research, Bhubaneswar, Odisha, India

Kaushik Mishra Department of Computer Science and Engineering, Sambalpur University Institute of Information Technology (SUIIT), Burla, India

Sanjoykumar Mishra Department of Electrical Engineering, G H Raisoni University, Amravati, India

Sarojananda Mishra Department of Computer Science Engineering and Applications, Indira Gandhi Institute of Technology, Sarang, Dhenkanal, Odisha, India

Sujogya Mishra Odisha University of Technology and Research, Bhubaneswar, India

Pratyasha Mohanty Rama Devi Women's University, Bhubaneswar, Odisha, India

Sagarika Mohanty National Institute of Technology, Rourkela, Rourkela, India

Subhadarshini Mohanty Odisha University of Technology and Research, Bhubaneswar, Odisha, India

Amritam Mohapatra Department of Mechanical Engineering, Indira Gandhi Institute of Techbology, Sarang, India

Subasish Mohapatra Odisha University of Technology and Research, Bhubaneswar, Odisha, India

R. Monisha Department of CSE, Global Academy of Technology, Bengaluru, Karnataka, India

S. Mousika Department of IT, Kongu Engineering College, Perundurai, India

Geetika Mudali Department of CSE, National Institute of Science and Technology Berhampur, Brahmapur, India

Kamalakanta Muduli Papua New Guinea University of Technology, Lae, Papua New Guinea

Manoj Kumar Muni Department of Mechanical Engineering, Indira Gandhi Institute of Technology, Sarang, Odisha, India

H. Muthukrishnan Department of IT, Kongu Engineering College, Perundurai, India

Sarmistha Nanda Department of Computer Science, Rama Devi Women's University, Bhubaneswar, India

G. K. Nayak CSE Department, International Institute of Information Technology, Bhubaneswar, Odisha, India

Smrutiranjan Nayak Department of Electrical Engineering ITER, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha, India

Chinmaya Ranjan Padhan Centre for Advanced Post Graduate Studies, Biju Patnaik University of Technology, Rourkela, Odisha, India

Ansuman Padhi Department of Mechanical Engineering, Indira Gandhi Institute of Technology, Sarang, Odisha, India

Rajalaxmi Padhy Odisha University of Technology and Research, Bhubaneswar, Odisha, India

Nayak Padmalaya Department of CSE, Gokaraju Lailavathi Womens Engineering College, Hyderabad, Telangana, India

M. G. Padmashree University Visvesvaraya College of Engineering, Bangalore University, Bengaluru, India;
Bangalore University, Bengaluru, India

Mrutyunjaya Panda Department of Computer Science and Application, Utkal University, Bhubaneswar, Odisha, India

Ribhu Abhusan Panda Department of Electronics and Communication Engineering, GIET University, Gunupur, Odisha, India

Sumanta Panda Department of Mechanical Engineering, VSSUT, Burla, Odisha, India

Chhabi Rani Panigrahi Department of Computer Science, Rama Devi Women's University, Bhubaneswar, India

S. Panigrahi School of Computer Engineering, KIIT Deemed to be University, Bhubaneswar, India

Punyaban Patel Department of Computer Science and Engineering, CMR Technical Campus, Kandlakoya, Hyderabad, India

Riyam Patel Department of Computer Science and Engineering (AI and ML), SRM Institute of Science and Technology, SRM University, Kattankulathur, Chennai, India

Bibudhendu Pati Department of Computer Science, Rama Devi Women's University, Bhubaneswar, India

Archana Patnaik Department of Computer Science Engineering, GIET University, Gunupur, Odisha, India

Binaya Kumar Patra Department of Computer Science Engineering and Application, Indira Gandhi Institute of Technology, Sarang, India

Madhusmita Patra Department of Computer Application, VSSUT, Burla, India

Prashanta Kumar Patra Odisha University of Technology and Research, Bhubaneswar, Odisha, India

Sanjay Kumar Patra Department of Computer Science Engineering and Application, Indira Gandhi Institute of Technology, Sarang, India

Sudhanshu Shekhar Patra School of Computer Applications, KIIT Deemed to be University, Bhubaneswar, India

Suresh Kumar Patra Institute of Physics, Sachivalaya Marg, Sainik School, Bhubaneswar, India;
Homi Bhaba National Institute, Mumbai, India

P. K. Pattnaik Odisha University of Technology and Research, Bhubaneswar, India

Arup Pradhan Department of Mechanical Engineering, VSSUT, Burla, Odisha, India

Pragyan Kumar Pradhan Department of Mechanical Engineering, Indira Gandhi Institute of Technology, Sarang, Odisha, India

Priyam Pradhan Department of Computer Science Engineering and Applications, Indira Gandhi Institute of Technology, Sarang, Dhenkanal, Odisha, India

Sateesh Kumar Pradhan Department of Computer Science, RBVRR Women's College, Hyderabad, India;
CS, Utkal University, Bhubaneswar, India

Sipali Pradhan Odisha University of Technology and Research, Bhubaneswar, India;
Department of Computer Application, Maharaja Sriram Chandra Bhanja Deo University, Baripada, Odisha, India

Jugge Praveen Department of EEE, GRIET, Hyderabad, Telangana, India

S. Preethi Department of Electronics and Communication Engineering, Kongu Engineering College, Erode, India

R. Priyadarshini Department of CS and IT, C.V. Raman Global University, Bhubaneswar, India

Alok Ranjan Prusty DGT, NSTI (W), RDSDE, Kolkata, West Bengal, India

Soumya Ranjan Prusty Institute of Management and Information Technology (IMIT) Cuttack, Cuttack, Odisha, India

Pratap Sekhar Puhan Department of Electrical and Electronics Engineering, Sreenidhi Institute of Science and Technology, Hyderabad, India

Rajiv Nandan Rai Subir Chowdhury School of Quality and Reliability, IIT Kharagpur, Kharagpur, West Bengal, India

S. Ramya Department of Electronics and Communication Engineering, Kongu Engineering College, Erode, India

R. Ramyea Department of Electronics and Communication Engineering, Kongu Engineering College, Erode, India

Shilpa Rana School of Physics and Materials Science, Thapar Institute of Engineering and Technology, Patiala, Punjab, India

Rajwant Singh Rao Guru Ghasidas Vishwavidyalaya (A Central University), Bilaspur, C.G., India

Kali Charan Rath GIET University, Gunupur, Odisha, India

Ashima Rout IGIT Sarang, Sarang, Odisha, India

Priyanka Rout Indira Gandhi Institute of Technology, Sarang, India

Diptendu S. Roy Department of Computer Science and Engineering, National Institute of Technology, Meghalaya, Shillong, India

Swarup Roy Network Reconstruction and Analysis (NetRA) Lab, Department of Computer Applications, Sikkim University, Gangtok, Sikkim, India

Abhaya Kumar Sahoo School of Computer Engineering, Kalinga Institute of Industrial Technology Deemed to be University, Bhubaneswar, India

Abinash Sahoo Department of Civil Engineering, National Institute of Technology Silchar, Assam, India

Amlan Sahoo Odisha University of Technology and Research, Bhubaneswar, Odisha, India

Anasuya Sahoo Talentsprint, Hyderabad, Telangana, India

Aradhana Sahoo Rama Devi Women's University, Bhubaneswar, Odisha, India

Bibhudatta Sahoo National Institute of Technology, Rourkela, Rourkela, India

Krushnashree Sushree Sangita Sahoo Department of Mechanical Engineering, Indira Gandhi Institute of Techbology, Sarang, India

Ramesh K. Sahoo CS, Utkal University, Bhubaneswar, India;
Indira Gandhi Institute of Technology, Sarang, Odisha, India

Satyabrata Sahoo Department of Electrical and Electronics Engineering, Nalla Malla Reddy Engineering College, Hyderabad, India

Anupama Sahu Department of Computer Science Engineering and Applications, Indira Gandhi Institute of Technology, Sarang, Dhenkanal, Odisha, India

Prashant Kumar Sahu Subir Chowdhury School of Quality and Reliability, IIT Kharagpur, Kharagpur, West Bengal, India

Bhaskar Sainath Jain University, Bangalore, Karnataka, India

G. Sandhya Department of Computer Science and Engineering, Sri Krishna College of Technology, Kovaipudur, Coimbatore, Tamilnadu, India

Deba Prakash Satapathy Department of Civil Engineering, Odisha University of Technology and Research, Bhubaneswar, Odisha, India

Sanjay Saxena CSE Department, International Institute of Information Technology, Bhubaneswar, Odisha, India

Snigdha Sen Department of CSE, Global Academy of Technology, Bengaluru, Karnataka, India

Rajiv Senapati Department of CSE, SRM University, Andhra Pradesh, India

J. Sengathir Department of Information Technology, CVR College of Engineering, Hyderabad, Telangana, India

Padmini Sethi RDW University, Bhubaneswar, Odisha, India

Purna Chandra Sethi Rama Devi Women's University, Bhubaneswar, Odisha, India

Srinivas Sethi CSEA, Indira Gandhi Institute of Technology, Sarang, Dhenkanal, India

Kabir Mohan Sethy Department of Geography, Utkal University, Bhubaneswar, Odisha, India

Mubashshirahbanu Shekh Department of Computer Science, Shrimad Rajchandra Institute of Management and Computer Application, Uka Tarsadia University, Gujarat, India

Borra Sivaiah Department of Computer Science and Engineering, CMR College of Engineering and Technology, Kandlakoya, Hyderabad, India

Kaushik Bhargav Sivangi School of Computer and Information Sciences, University of Hyderabad, Hyderabad, India

K. S. Sri Lakshmi Department of CSE, Global Academy of Technology, Bengaluru, Karnataka, India

Adunuri Srilekha Electrical and Electronics Engineering, Sreenidhi Institute of Science and Technology, Hyderabad, India

Nirmal Keshari Swain Gandhi Institute for Education and Technology, Baniatangi, Khurda, Odisha, India

Sarat Chandra Swain School of Electrical Engineering, KIIT University, Bhubaneswar, Odisha, India

Binon Teji Network Reconstruction and Analysis (NETRA) Lab, Department of Computer Applications, Sikkim University, Gangtok, India

Subhranshu Sekhar Tripathy Department of Computer Science and Engineering, National Institute of Technology, Meghalaya, India

Siba K. Udgata School of Computer and Information Sciences, University of Hyderabad, Hyderabad, India

P. B. Ullagaddi S.G.G.S. Institute of Engineering and Technology, Nanded, India

Trivedi Veena Computer Engineering Department, VESIT, Mumbai, India

K. R. Venugopal University Visvesvaraya College of Engineering, Bangalore University, Bengaluru, India;
Bangalore University, Bengaluru, India

GKEAE: Group Key Exchange and Authentication with ECC in Internet of Things

M. G. Padmashree, Mallikarjun, J. S. Arunalatha, and K. R. Venugopal

Abstract The Internet of things (IoT) is composed of smart devices that collaborate to provide complex services. The security providers forcefully reorganize existing capabilities to address IoT security risks. The sensor deployment in deregulated context and heterogeneous devices, applications, and communication models complicates the decryption of devices and broaden the opening for an intruder. An authentic device delivers malicious data when an adversary controls its logical or actual context. The proposed Group Key Exchange and Authentication with Elliptic Curve Cryptography (ECC) in the Internet of things achieve secure key distribution and enhances security. The IoT device authentication uses ECC when an IoT device enters or exits the group. The serviceability of IoT devices increases by integrating access authentication and data transmission. The GKEAE achieves a lower computation time of the group key distribution than the fast authentication scheme.

Keywords Authentication · Elliptic curve cryptography · Group key distribution · Group key management · Internet of things · Re-keying

1 Introduction

Internet of things (IoT) is a breakthrough technological innovation in information and communication technology (ICT) [17]. It extends the preexisting whenever-wherever to connect whatever concept. According to various information technology industries, there is a swift movement toward the practical implementation of IoT. IoT includes things and communication networks. The primary function of the IoT devices, directly and indirectly, associates with ICT, e.g., microwave oven. The communication networks identify, resolve, and integrate the things that capture the process and learn the data. In the IoT, interconnected devices share information and resources to accomplish a specific functionality [16]. The heterogeneity of

M. G. Padmashree (✉) · Mallikarjun · J. S. Arunalatha · K. R. Venugopal
University Visvesvaraya College of Engineering, Bangalore University, Bengaluru, India
e-mail: padmashree.mg@gmail.com

Bangalore University, Bengaluru, India

IoT devices and technology improves the efficiency of applications, maximizing the benefits for the user. Adopting new technology must take into account the new and challenging elements involved. The increasing progress of the IoT needs to address deployment-related security issues.

In a group interaction framework, devices generate and deliver messages to all the group nodes *via,* broadcasting or multicasting to increase efficiency. In IoT, the applications with data sharing, interaction, storage, and device-to-device messaging necessitate multicasting data. Group communication security provides confidentiality, authentication, and trustworthiness of the inter-group data using suitable cryptographic algorithms without disturbing the multicast data flow. An efficient and scalable security achievement is hard to develop as devices enter and exit the group.

Various devices consistently attempt to contact the IoT devices, indicating usage of device power even on inactive core functionality. Therefore, billions of these devices are concerned about backup energy overhead, despite the low power requirements of the individual devices. However, authenticating individual IoT devices demands an extended computation cost as the system consists of ample devices. Group authentication allows all users to authenticate simultaneously. Public key cryptography is an approach of assurance to overcome data privacy issues [20]. But standard algorithms are complex and hence cannot incorporate in IoT applications [1]. A trusted authority [22] using the extended Euclidean algorithm manages the groups and generates the group keys [12]. Asymmetric univariate/bivariate polynomial token and exclusive-OR encryption provide authentication in the group [4]. The security using elliptic curve cryptography (ECC) is higher than RSA since none of the competent algorithms breaks the discrete logarithm problem in polynomial time [19]. ECC requires 12 times smaller secret keys for the same security level with reduced processing time and storage.

Motivation: A membership modification must follow a pre-shared cryptographic key updation to avoid accessing further group data by the exited device [21] and past communications are inaccessible to exited devices. Group communication using asymmetric cryptographic encryption achieves mutual authentication, integrity, and confidentiality, and multicast symmetric cryptographic data encryption provides scalability, reducing complexity [3]. The group members share the symmetric group key among them. Group Key Exchange and Authentication with ECC (GKEAE) authenticate IoT devices to transmit by secure group key.

Contributions: The primary contributions in the proposed Group Key Exchange and Authentication with ECC in the Internet of things are:

 (i) Providing secure key distribution and enhancing security by authentication on device entry or exit to the group.
(ii) Decreasing the computation time of inter- and intra-group key distribution using ECC integrating access authentication and data transmission.

Organization: The paper is structured as follows: Sect. 2 summarizes the related works of the Group Key Exchange and Authentication in IoT, Sect. 3 abstracts the background work. Section 4 describes the proposed Group Key Exchange and

Authentication with ECC for securing data communication in IoT. Section 5 presents the performance analysis of the proposed GKEAE. Section 6 consolidates the work with the conclusions.

2 Related Works

Group authentication with a fault tolerance scheme provides scalability [5] using Steiner Oval Combinatorics. IoT devices perform appropriately in the absence of few group members allowing inter-group key sharing. The fault tolerance relies on the threshold, i.e., the several group elements [6] are flexible, scalable, and resilient. Zhang et al. [23] introduced a cross-layer authentication framework to achieve earlier and accurate authentication dynamically. Multiple attributed physical layer authentication (PLA) provides authentication with reliability and high accuracy. PLA in dynamical framework experiences a decline in accuracy; the higher layer authentication guides PLA to adjust parameters. PLA effectively rectifies the performance drop, satisfying authenticate accuracy and time in dynamic networks.

An identity-based dynamically authenticating group key agreement scheme [10] refreshes group keys on association change in one iteration. It is resistant to controlled key attacks providing forward confidentiality. The user messages sent for authentication are more. Kumar et al., [9] designed a bilinear pair-free group key distribution using elliptic curve operations, Lagrange interpolation, and bottom-up access hierarchy. After verifying the predetermined features and not the identity, a session establisher securely transfers the group key to every genuine member using polynomial interpolation, elliptic curve scalar point multiplication. It is resilient to collusion attack due to common polynomial set definition.

Parne et al. [15] designed a group AKA protocol for communication in IoT. Symmetric key cryptography and cumulative MAC parallelly authenticate group devices. Protecting symmetric shared key between device and server and ensuring the confidentiality of device during the authentication solve the unique key problem. The scheme maintains non-linkage, traceability, and bulk device forward and backward key secrecy. A session key integrated between the member and the leader secures the group device communication. The unique identity is pointing to the shared secret between device and server updated after each successful authentication. A third party generates the group key and the shared secret. An insecure path between router and server is ignored.

Mahmood et al. [11] presented distributed multi-party session key agreement protocol using chaotic map-based Hash and Chebyshev polynomial integrated cryptographic operations. Group head computes and sends polynomial to server. A new polynomial computation with partial group attributes authenticates the group head. The server generates and shares the session key with communicating nodes. The authentication protocols cluster-based [14] use Hashing [8], the inter-network compatible [3, 7].

3 Background Work

The LEACH [18] clustering algorithm distributes group data using local processing with reduced global communication. The random rotation of the group admin extends the system availability. Each non-Admin node n in a set S acquires a random possibility R between 0 and 1. The first node below threshold $Tr(n)$ in an Ith iteration is a group admin given in Eq. (1).

$$\text{Tr}(n) = \begin{cases} \frac{R}{(1-R*(I \mod \frac{1}{R}))} & \text{if } n \in S, \\ 0 & \text{if } n \notin S \end{cases}$$ (1)

Group admin broadcast messages to all nodes. The nodes inform the group admin to join its group. Group admin creates a request-based schedule and informs the time slot for transmission from the node.

The group cardinality influences the protocol performance that is inversely proportional to the range to be covered, i.e., few nodes far from admin and consumes more energy. The group admin and the router communication increases the transmission cost [2, 14].

3.1 Problem Statement

For a set of given IoT devices, to design a Group Key Exchange and Authentication scheme for IoT applications, the objectives of the proposed GKEAE work are:

(i) To enhance security by secure key distribution and authentication on device entry or exit to the group.
(ii) To extend serviceability by decreasing the computation time of inter- and intra-group key distribution using ECC integrating access authentication and data transmission.

4 The Proposed Group Key Exchange and Authentication with ECC in Internet of Things

The Group Key Exchange and Authentication in IoT with elliptic curve cryptography is proposed to ensure secure data communication.

4.1 System Architecture

The architecture of Group Key Exchange and Authentication with ECC is shown in Fig. 1. Sending and receiving group members effectively encrypt and decrypt the message using elliptic curve cryptography. The nodes cluster into groups in the network. Each group consists of a group admin. The group admin comprises more computation power compared to other nodes in the group. The group admin is used for inter-group communication to efficiently make the overall network energy efficient and use the energy of each node.

4.2 System Block

The block diagram of the Group Key Exchange and Authentication with ECC (GKEAE) is given in the following Fig. 2.

The system initialization module configures the elliptic curve parameters. The group key exchange module generates the group key involving a set of group members and distributes among them. The authentication of the devices *via* the group admin by the server secures and enhances data transmission.

4.3 The Proposed Algorithm

The proposed Group Key Exchange and Authentication with ECC (GKEAE) ensures secure IoT data communication.

Initialization: IoT devices register to the group registered to the server *via* the gateway using the LEACH threshold given in Eq. (1) and forms the group using the threshold

Fig. 1 Architecture of the proposed GKEAE

Fig. 2 Architecture of the
proposed GKEAE
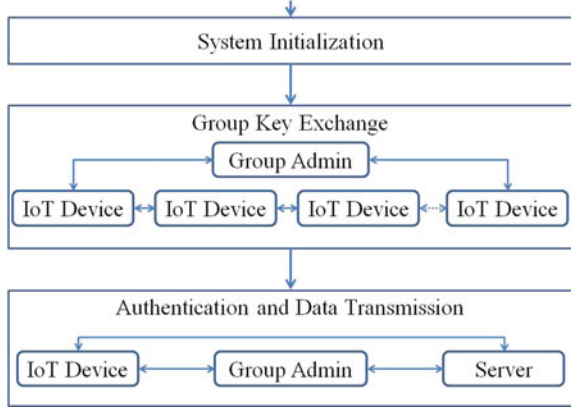
given in Eq. (2) for the node Di node where Di_r is the success probability to be a
group admin in jth iteration.

$$Di_{Tr} = \frac{Di_r}{(1 - Di_r * (j \mod \frac{1}{Di_r}))} \tag{2}$$

The system parameters of ECC configure for an authentication process.

Group Key Exchange: Group admin sends the Seed Key $K0_{\text{grp}} = dG$ to neighboring
group member. Every group member D_i computes Ki_{pvt} and sends it to the neighbor
device. Group admin generates a Group Initialization Key, $K0_{\text{Intrcd}} = dG$. $D1$ gen-
erates a Private Key a, $K1_{\text{pvt}}$ and computes Public Key $K1_{\text{grp}} = aG$. $D2$ generates a
Private Key b, $K2_{\text{pvt}}$ and computes Public Key $K2_{\text{grp}} = bG$. $D3$ generates a Private
Key c, $K3_{\text{pvt}}$ and computes Public Key $K3_{\text{grp}} = cG$. The devices multicast the public
keys within the group given in Eq. (3)

$$\forall_{i=1}^{n} Ki_{\text{grp}} \rightarrow \forall_{j=1, j!=i}^{n} D_j \tag{3}$$

Group admin shares $K0_{\text{Intrcd}} = dG$ to $D3$. $D1$ computes $K01_{\text{Intrcd}} = cdG$ and
sends to $D2$. $D2$ computes $K012_{\text{Intrcd}} = bcdG$ and sends to $D3$. $D3$ computes
$K0123_{\text{Intrcd}} = abcdG$ [13] and sends to group admin. Group key generating Devices
D_i, for $i = 1$ to n, forwards the intercede keys to the next device which is in Eq. (4).

$$\forall_{i=1}^{n} Ki_{\text{Intrcd}} \rightarrow D_{i+1} \tag{4}$$

Group admin receives the Key $abcdG$. Group admin updates the Group Key
$K_{\text{grp}} = dabcdG$ in Eq. (5) for $j = i - 1$ and $n + 1$th device is the Admin.

$$\forall_{i=1}^{n} Kij_{\text{Intrcd}} = d \Pi Ki_{\text{pvt}} Kj_{\text{Intrcd}} \tag{5}$$

Authentication and Data Transmission: The group admin authenticates the IoT device *via* the server. The authenticated sending IoT device encrypts the data using Group Key $K_{\text{grp}} = Kn0_{\text{Intrcd}}$. The authenticated receiving IoT fevice with $Kn0_{\text{Intrcd}}$ decrypts data to complete the data communication.

The steps involved in Group Key Exchange and Authentication with ECC (GKEAE) are given in Algorithm 1 that includes initialization (Steps 1–3), group key exchange (Step 4–13), authentication and data transmission (Step 14–21).

Algorithm 1 $GKEAE$; Group Key Exchange and Authentication with ECC

Input: Set of Devices
Output: Secure Data Communication with Authentication
  Group formation by LEACH algorithm using Eqs. (1) and (2)
  IoT devices register to the Group registered to the Server *via* the Gateway
  Initialize the System Parameters of Elliptic Curve Cryptography
  repeat
    Group Admin sends the Seed Key $K0_{\text{Intrcd}}$ to neighboring Group Member
    for all D_i in Group Members **do**
      Compute Ki_{Intrcd} and send to neighbor device using Eq. (4)
    end for
    Group Admin updates Group Key $K_{\text{grp}} = Kn0_{\text{Intrcd}}$ using Eq. (5)
    if new IoT Device Enters/Exits the Group **then**
      ReKeying Triggered by Group Admin and Distributed along with the new / remaining IoT Device
    end if
  until all Groups obtain the Group Keys
  The Group Admin Authenticates the IoT device with ECC Key *via* the Server
  if *valid* **then**
    Sending IoT device Encrypts the data using Group Key K_{grp}
  end if
  The Group Admin Authenticates the IoT device *via* the Server.
  if *valid* **then**
    Receiving IoT Device with Group Key K_{grp} Decrypts data.
  end if

---

5 Performance Analysis

The proposed GKEAE for secure IoT device communication is implemented using Java. The performance of GKEAE is compared with the fast authentication scheme (FAS) [3]. GKEAE simplifies the authentication process and alleviates the network load, ensures a fast protective barrier *viz.,* anonymousness and non-repudiation. The proposed GKEAE can resist various security attacks efficiently.

Computation Time: The computation time of GKEAE is compared with FAS which is shown in Figs. 3 and 4. GKEAE takes 51 ms for computation time, whereas FAS takes 945.36 ms for 100 nodes. With a group size of 5, GKEAE takes 98 ms for computation, whereas FAS takes 976.48 ms. In a group size of 10, GKEAE takes 152 ms for
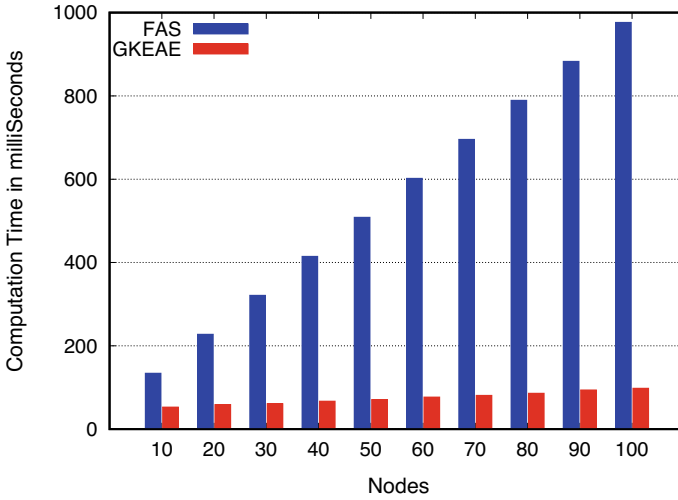
Fig. 3 Computation time of GKEAE ($m = 5$)
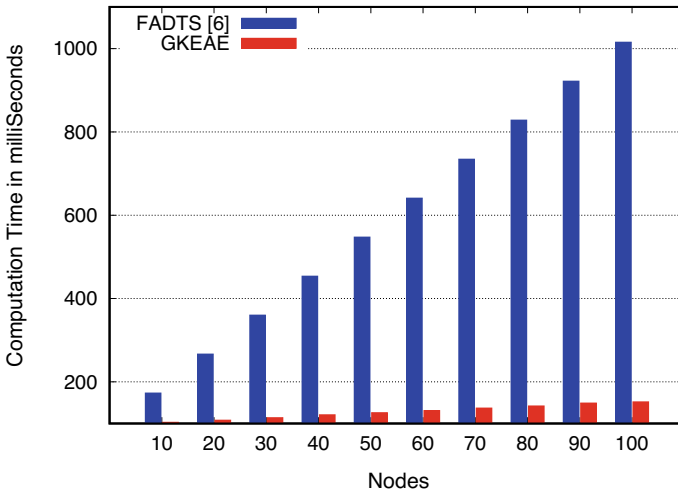
Fig. 4 Computation time of the proposed GKEAE ($m = 10$)

computation time, whereas FAS takes 1015.38 ms. The reduction in computation time is the result of ECC use for authentication. The approximate computation time is given by Eq. (6) for n nodes and m group size.

$$\frac{3nm}{250} + \frac{11n}{25} + 10m - 3 \tag{6}$$

GKEAE provides robust protection with ideal efficiency for IoT applications. The intra- and inter-group systems allow the nodes to transmit data, reducing transmission time with less hop count. GKEAE shows five times decrease in the computation time compared to the FAS due to ECC integrated access and authentication. Large groups require large bandwidth because passing around partial keys in the intermediate rounds of computation.

Security Analysis: The subexponential algorithms for factoring and only fully exponential algorithms for ECC provide smaller bit lengths to achieve higher security level. Each group establishes a group key in the combined sequential ECDH and gradually build up the shared key by multiplying in the secret keys sequentially across multi-rounds. The group admin updates the keys on group cardinality variation for security concerns *viz.,* MiTM, replay, and DoS attacks.

6 Conclusions

The Internet of things (IoT) identifies and communicates with smart devices. The scope of IoT applications is extensive and accordingly connects a large set of devices. These devices can smartly perform aggregate, evaluate, and determine without human interference where security is the primary requisite; specifically, authentication is the requirement to overcome the exposure of a device in the IoT environment. The proposed GKEAE enhances security and extends the serviceability of IoT devices using secure group key distribution and authentication when an IoT device enters or exits the Group. The computation time key distribution using ECC integrating access authentication and data transmission is five times less compared to the fast authentication scheme. The use of edge servers in preference to the group admin computation can increase the network performance.

References

1. Aydin Y, Kurt GK, Ozdemir E, Yanikomeroglu H (2020) A flexible and lightweight group authentication scheme. IEEE J Internet Things 7(10):1–11
2. Bhola J, Soni S, Cheema GK (2020) Genetic algorithm based optimized leach protocol for energy efficient wireless sensor networks. J Amb Intell Human Comput 11(3):1281–1288
3. Cao J, Yu P, Ma M, Gao W (2019) Fast authentication and data transfer scheme for massive NB-IoT devices in 3GPP 5G network. IEEE J Internet Things 6(2):1561–1575
4. Cheng Q, Hsu C, Harn L (2020) Lightweight noninteractive membership authentication and group key establishment for WSNs. Hindawi J Math Probl Eng 2020:1–9
5. El Mouaatamid O, Lahmer M, Belkasmi M (2020) A scalable group authentication scheme based on combinatorial designs with fault tolerance for the Internet of things. SN Comput Sci 1(4):1–13
6. Elmouaatamid O, Lahmer M, Belkasmi M (2017) Group authentication with fault tolerance for Internet of things. Lecture notes in computer science, vol 10542, pp 299–307
7. Jiang X, Lora M, Chattopadhyay S (2020) An experimental analysis of security vulnerabilities in industrial IoT devices. ACM Trans Internet Technology 20(2), 16:1–16:24

8. Kavianpour S, Shanmugam B, Azam S, Zamani M, Narayana Samy G, De Boer F (2019) A systematic literature review of authentication in Internet of Things for heterogeneous devices. Comput Netw Commun **2019**

9. Kumar A, Verma R (2020) Attribute-based authenticated group key transfer protocol without pairing. Wireless Pers Commun 113(4):1791–1805

10. Li F, Xie D, Gao W, Wang XA, Yan J (2016) An ID-based dynamic authenticated group key agreement scheme with optimal round complexity from pairings. In: Proceedings of the tenth IEEE international conference on complex, intelligent, and software intensive systems (CISIS), pp 468–472

11. Mahmood Z, Ullah A, Ning H (2018) Distributed multiparty key management for efficient authentication in the Internet of things. IEEE Access 6:29460–29473

12. Mansour A, Malik KM, Alkaff A, Kanaan H (2021) ALMS: asymmetric lightweight centralized group key management protocol for VANETs. IEEE Trans Intell Transp Syst 22(3):1–16

13. Padmashree MG, Khanum S, Arunalatha JS, Venugopal KR (2021) ETPAC: ECC based trauma plight access control for healthcare Internet of things. Int J Inf Technol 13(4):1481–1494

14. Padmashree MG, Ranjitha, Arunalatha JS, Venugopal KR (2020) CKDAC: cluster-key distribution and access control for secure communication in IoT. In: Seventh IEEE Uttar Pradesh section international conference on electrical, electronics and computer engineering, pp 1–6, Nov 2020

15. Parne BL, Gupta S, Chaudhari NS (2018) SEGB: security enhanced group based AKA protocol for M2M communication in an IoT enabled LTE/LTE—a network. IEEE Access 6:3668–3684

16. Qiu J, Fan K, Zhang K, Pan Q, Li H, Yang Y (2019) An efficient multi-message and multi-receiver signcryption scheme for heterogeneous smart mobile IoT. IEEE Access 7:180205–180217

17. Tewari A, Gupta BB (2020) Security, privacy and trust of different layers in Internet-of-things (IoTs) framework. J Fut Gen Comput Syst 108:909–920

18. Varshney S, Kuma R (2018) Variants of LEACH routing protocol in WSN: a comparative analysis. In: Eight IEEE International conference confluence 2018 on cloud computing, data science and engineering, pp 199–204

19. Wu ZY (2019) Group-oriented cryptosystem for personal health records exchange and sharing. IEEE Access 7:146495–146505

20. Xu P, He S, Wang W, Susilo W, Jin H (2018) Lightweight searchable public-key encryption for cloud-assisted wireless sensor networks. IEEE Trans Ind Inf 14(8):3712–3723

21. Yang Z, He J, Tian Y, Zhou J (2020) Faster authenticated key agreement with perfect forward secrecy for industrial Internet-of-things. IEEE Trans Ind Inf 16(10):6584–6596

22. Zhang J, Zhong H, Cui J, Xu Y, Liu L (2020) An extensible and effective anonymous batch authentication scheme for smart vehicular networks. IEEE J Internet Things 7(4):3462–3473

23. Zhang Z, Li N, Xia S, Tao X (2020) Fast cross layer authentication scheme for dynamic wireless network. In: Proceedings of the IEEE wireless communications and networking conference (WCNC), pp 1–6

PIDA Regulator for Frequency Limitation of Conventional Power Systems

Smrutiranjan Nayak, Sanjeeb Kumar Kar, Subhransu Sekhar Dash, Madhab Chandra Das, and Sarat Chandra Swain

Abstract PIDA regulator manages automatic generation control of power installation. In addition, in the affectability examination, the system limits, burden and the region of disturbance are changed, and the results are taken apart. PIDA is introduced to fulfill details for transient and consistent state reaction of a third request control framework. The proposed controller design techniques are also employed for controlling AC motor system models. In the proportional controller, rectification is applied to the controlled variable which is relative to the distinction between the ideal worth and estimated esteem. The integral controller can eliminate the steady-state error. The subordinate regulator detects the pace of progress of the blunder signal. Speed increase regulator is a high request control alternative that adds high request gains to the position control calculation. The presentation from the affectability evaluation appropriateness of hSGA/PS tuned PIDA regulator on AGC of the power structure.

Keywords Multi-area power system · Hybrid search group algorithm (hSGA) · PIDA controller · Frequency

S. Nayak (✉) · S. K. Kar
Department of Electrical Engineering ITER, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha, India
e-mail: Smrutikiit40@gmail.com

S. K. Kar
e-mail: sanjeebkar@soa.ac.in

S. S. Dash
Department of Electrical Engineering, Government College of Engineering, Keonjhar, Odisha, India
e-mail: subhransudash_fee@gcekjr.ac.in

M. C. Das · S. C. Swain
School of Electrical Engineering, KIIT University, Bhubaneswar, Odisha, India
e-mail: scswainfel@kiit.ac.in

1 Introduction

In three locale organizations, units of the individual region are compared with the different regions by techniques for power lines. An exertion is always made for the development of the PIDA regulator with AGC for which it is adjusted by hSGA/PS strategy [1–3]. An example of search is utilized to the unavoidable results of SGA adjusted PIDA respects. Thus, warm arrangement of various cutoff focuses with nonlinearity is always considered for appraisal. Finally, to favor the sufficiency of the presented arrangement approach, affectability assessment is done under contrasted system limits and the working weight. The amendment is applied to the controlled variable which is corresponding to the contrast between the ideal worth and estimated esteem. An integral controller can eliminate the steady-state error that occurs with a proportional controller [4–6]. The regulator detects the pace of progress of blunder signal and contributes a segment of the yield signal that is corresponding to a subordinate of the mistake signal. It is a high request control choice that adds high request gains to the position control calculation. Power framework security is improved by utilizing a programmed voltage controller to change the voltage of the terminal of the coordinated generator. PIDA and PID regulators are utilized in AVR for excitation control. PIDA regulator is superior to PID regulator [7–9].

2 System Analysis and Objective Function

In Fig. 1, a three-region structure is shown in which three warm units of different
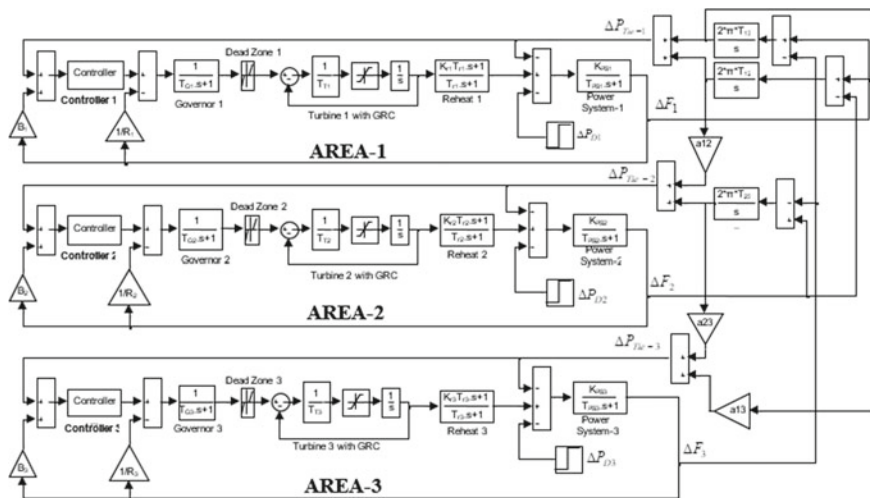
Fig. 1 Three region tests

requirements of each area are displayed below. The design of PIDA regulator is shown in Fig. 2 [10, 11].

The target work is given by

$$J = \text{ITAE} = \int_0^{t_{\text{sim}}} (|\Delta F_i| + |\Delta P_{\text{Tie}-i}|) \cdot t \cdot dt \tag{1}$$

where t_{sim} is the simulation time, $\Delta F_i =$ deviation in frequency, and $\Delta P_{\text{tie}} =$ deviation in tie-line power.

In PID controller, K_p reduces time rise, K_i improves steady-state error, and K_d improves stability margin and reduces overshoot.

The Laplace function PID controller is given by

$$G(s) = K_p + K_i/s + K_d S \tag{2}$$

In PIDA, regulator is the same as the PID regulator, yet it has one extra increase boundary which is speed increase acquire (K_A) as it is the third request framework and expansion channel component for improvement of steadiness. PIDA controller is expressed below

$$G(S) = K_p + K_i/s + K_d S/(S+d) + K_a S^2/(S+d)(S+e) \tag{3}$$

So, d and e are filter elements.

Fig. 2 Shape of PIDA controller
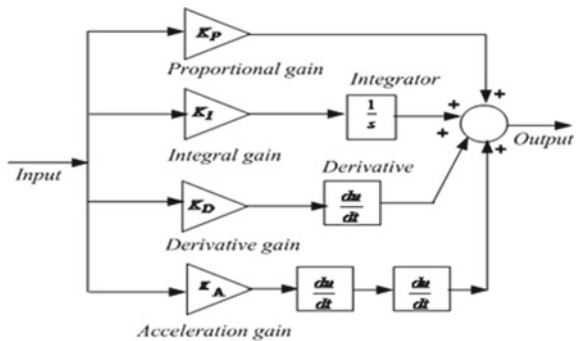
3   Search Group Algorithm

Different steps of SGA are explained as follows.

3.1   Beginning Population

It is given by

$$P_{ij} = X_j{}^{\min} + \left(X_j{}^{\max} - X_j{}^{\min}\right) U(0, 1)$$

where j $= 1,\ldots n$ and $i = 1,\ldots n_{\text{pop.}}$

P_{ij} is jth parameter of ith individual population P, and $X_j{}^{\min}$ and $X_j{}^{\max}$ are minimum and maximum limits.

3.2   Initial Search Bunch Choice

Every individual populace's targeted work is resolved, and from population P, a pursuit bunch R is focused.

3.3   Change of the Search Bunch

$X_j{}^{\text{mut}} = E[R_j] + t \in \sigma[R_j]$ for $j = 1, \ldots n$, so $X_j{}^{\text{mut}}$ is mutated individual of jth parameter and $t =$ control parameter, $\in =$ random number, $E =$ mean operator, $\sigma =$ standard deviation, and $R_j = j$th row of R.

3.4   Formation of Groups of Individual Hunt Bunch Part

$$X_j{}^{\text{new}} = R_{ij} + \alpha \in \text{ for } j = 1, \ldots n.$$

So, perturbation size is controlled by α.

$$\alpha^{k+1} = b\,\alpha^k$$

Here, b is a parameter.

3.5 Plan for New Bunch

$it^{\max}$ iterations, best part from every family, are considered to form new pursuit bunch. When the emphasis number is more prominent, then new inquiry bunch is made by taking fittest focuses from every one of the families.

4 Pattern Search

Pattern search calculation equivalent to string looking through calculation is also known as direct search. Target capacities are figured at network focuses in the grouping it has been made. The example inquiry improvement routine is a subordinate free advancement strategy that is appropriate to tackle an assortment of streamlining issues. The example search calculation continues by registering a grouping of focuses that could conceivable provide ways to deal with the ideal point. The example search starts at the underlying point X0 that is given as a beginning stage by the client [12, 13].

The example search calculation continues by processing an arrangement of focuses that might possible provide ways to deal with the ideal point. The calculation begins by building up a bunch of focuses called network. The model inquiry starts at the initial point X0 that is mentioned as an early phase by client. At the main accentuation, with a scalar equal to 1 called network size, the model vectors are worked as [−1 0], [0 −1], [1 0] and [0 1], and they may be called as course vectors. The example search calculation stops when any of the accompanying conditions happens.

1. The cross-section size is not as much as lattice resilience.
2. The quantity of emphases performed by the calculation arrives at the worth of max. iteration.
3. The all-out number of target work evolutions performed by the calculation arrives at the worth of max work assessments.
4. The adjustment of the target work from one fruitful survey to next effective survey is not exactly the target work resistance.

5 Analysis of Result

The best interesting execution is refined of hSGA/PS adjusted PIDA controller, basic improvement is seen with stated SGA adjusted PIDA regulator than SGA stream-lined PID regulator, and additionally, SGA improved PID regulator gives supported reaction over FA refreshed PID regulator. The frame structure implementation for a concurrent 10% advancement inconvenience arrangement in regions 1 and 2 with hSGA/PS adjusted PIDA regulator limits under structure cutoff points and works on weight condition groups are introduced here [14–20] (Fig. 3).
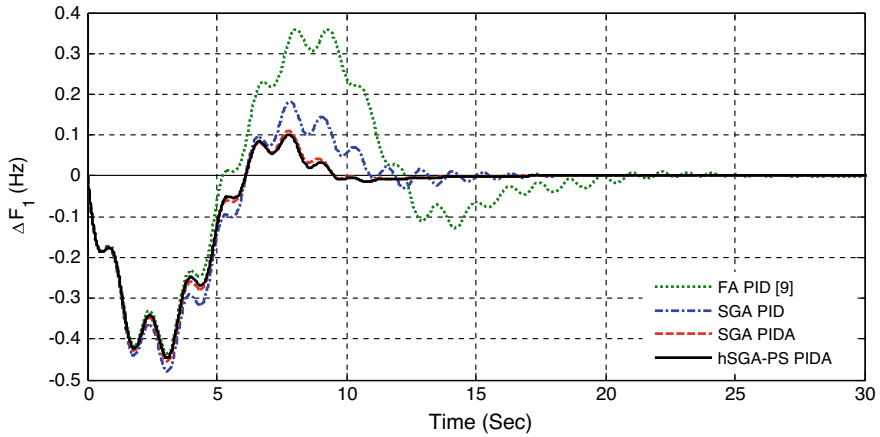
Fig. 3 Frequency deviance of zone-1 for simultaneous 10% SLD in zone-1 and zone-2

6   Conclusion

SGA utilizes throughout here to upgrade the PIDA/PID regulators using ITAE target limits. By then, PIDA regulators are adjusted by SGA, and the outcome uncovers that SGA improved PIDA regulator. To abuse region, search limits of model inquiry, pattern search is utilized to adjust the PIDA regulator limit.

The advantages of hSGA/PS adjusted PIDA regulator over SGA adjusted PIDA regulator, SGA adjusted PID regulator and FA adjusted PID regulator are demonstrated. The suggested hSGA/PS-based PIDA regulator gives improved execution that showed up diversely according to SGA adjusted PIDA regulator.

References

1. Kundur P (2009) Power system stability and control, 8th edn. Tata Mc-Graw Hill, New Delhi
2. Mishra S, Mallesham G, Sekhar PC (2013) Biogeography based optimal state feedback controller for frequency regulation of a smart micro grid. IEEE Trans Smart Grid 4:628–637
3. Shabani H, Vahidi B, Ebrahimpour MA (2012) Robust PID controller based on imperialist competitive algorithm for load-frequency control of power systems. ISA Trans 52:88–95
4. Sahu RK, Panda S, Rout UK, Sahoo DK (2016) Teaching learning-based optimization algorithm for automatic generation control of power system using 2-DOF PID controller. Int J Electr Power Energy Syst 77:287–301
5. Sivalingam R, Chinnamuthu S, Dash S (2018) A modified whale optimization algorithm based adaptive fuzzy-logic PID controller for load frequency control of autonomous power generation systems. J Control Measure Electron Comput Commun 58(4):410–421
6. Saikia LC, Nanda J, Mishra S (2011) Performance comparison of several classical controllers in AGC for multi-area interconnected thermal system. Int J Electr Power Energy Syst 33:394–401

7. Sahu RK, Panda S, Padhan S (2015) A hybrid firefly algorithm and pattern search technique for automatic generation control of multi area power systems. Int J Electr Power Energy Syst 64:9–23
8. Rajesh KS, Dash SS (2019) Load frequency control of autonomous power system using adaptive fuzzy based PID controller optimized on improved sine cosine algorithm. J Ambient Intell Humaniz Comput 10(6):2361–2373
9. Abd-Elazim SM, Ali ES (2018) Firefly algorithm-based load frequency controller design of a two area system composing of PV grid and thermal generator. Electr Eng 100(2):1253–1262
10. Sahu BK, Mohanty PK, Panda S, Kar SK, Mishra N (2012) Design and comparative performance analysis of PID controlled automatic voltage regulator tuned by many optimizing liaisons. In: International Conference on Advances in Power Conversion and Energy Technologies (APCET), pp 1–6. IEEE (2012)
11. Bao Y, Hu Z, Xiong T (2013) A PSO and pattern search based memetic algorithm for SVMs parameters optimization. Neurocomputing 117:98–106
12. Padhan S, Sahu RK, Panda S (2014) Application of firefly algorithm for load frequency control of multi-area interconnected power system. Elect Power Comp Syst 42:1419–1430
13. Sornmuang S, Sujitjorn S (2010) GA-based PIDA control design optimization with an application to ac motor speed control. İnt J Math Comput Simul 4:67–80
14. Puangdownreong D (2012) Application of current search to optimum PIDA controller Design. Intell Control Autom 3:303–312
15. Padhi JR, Debnath MK, Pal S, Kar SK (2019) AGC investigation in wind-thermal-hydro-diesel power system with 1 plus fractional order integral plus derivative controller. Int J Recent Technol Eng 8(1):281–286
16. Mohanty PK, Sahu BK, Pati TK, Panda S, Kar SK (2016) Design and analysis of fuzzy PID controller with derivative filter for AGC in multi-area interconnected power system. IET Gener Transm Distrib 10(15):3764–3776
17. Nayak S, Kar SK, Dash SS (2021) Combined fuzzy PID regulator for frequency regulation of smart grid and conventional power systems. Indonesian J. Electr Eng Comput Sci 24(1):12–21
18. Nayak S, Dash SS, Kar SK (2021) Frequency regulation of hybrid distributed power systems integrated with renewable sources by optimized type-2 fuzzy PID controller. In: 9th International Conference on Smart Grid, icSmartGrid, pp 259–263
19. Swain SC, Mohapatra S, Panda S, Nayak SR (2012) Real-coded genetic algorithm for robust design of UPFC supplementary damping controller. Int J Power Syst Oper Energy Manage 1(3):53–58. ISSN-2231-4407
20. Padhi JR, Debnath MK, Kar SK, Sahu BK (2020) LFC analysis in PV-thermal system using non-integer controller. Lect Notes Electr Eng 630:475–484

Synchronization and Its Use in Communication Network with Frequency Control

Smrutiranjan Nayak, Sanjeeb Kumar Kar, Subhransu Sekhar Dash, and Madhab Chandra Das

Abstract   In this paper, another fragmentary request turbulent framework containing a few nonlinearity terms is presented. This new framework can energize stowed away turbulent attractors or self-invigorated tumultuous attractors relying upon the picked framework boundaries or its part request. A few elements of this new framework, like tumultuous attractors, harmony focuses, are examined logically and mathematically. Then, at that point, versatile control laws are created for accomplishing confusion synchronization of two indistinguishable new frameworks with dubious boundaries: one of these two new indistinguishable frameworks is the expert and the other is the slave. Moreover, in disarray application fields, these expert and slave synchronized frameworks are applied in secure correspondence to go about as transmitter and beneficiary individually. Mathematical test outcomes showed the chance of utilizing this proposed fragmentary request tumultuous framework in high-security correspondences.

S. Nayak (✉) · S. K. Kar
Department of Electrical Engineering ITER, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha, India
e-mail: Smrutikiit40@gmail.com

S. K. Kar
e-mail: sanjeebkar@soa.ac.in

S. S. Dash
Department of Electrical Engineering, Government College of Engineering, Keonjhar, Odisha, India
e-mail: subhransudash_fee@gcekjr.ac.in

M. C. Das
School of Electrical Engineering, KIIT University, Bhubaneswar, Odisha, India

1 Introduction

Recent research has concentrated on chaotic systems with hidden attractors. It is very important in many application fields of science such as induction motors for drilling, bridge wings design and secure communication schemes [1, 2]. It exhibits more complex dynamical behaviour than their integer analogues and thus plays an important role in fields of secure communication applications and high-level cryptography. Also, however the fractional order (q) of the system can be changed in (0, 1) range that gives an additional system parameter as well as the original system parameters [3, 4]. Through the development and applications of chaos theory, many fractional-orders chaotic systems have been introduced, such as fractional-order Lorenz system, fractional-order Chen system, fractional-order Liu chaotic system, fractional-order financial systems and various other systems [5, 6]. Various regulators such as sliding mode control, adaptive control, active control, passive control, impulsive control and so on as in [7, 8]. In the secure communication application field, the synchronization is achieved between two chaotic systems: the first one is a master and the second response one is a slave, where the master presents the transmitter side, and the slave acts the receiver side. The researchers employed many fractional-order chaotic systems, and they used in secure communication and encryption [9, 10]. As the new system is being used in a secure communication system, an adaptive synchronization mechanism has been established between two identical new systems, one of which is acting as the master (transmitter) and the other as the slave (receiver) [11–13]. Our work, tests and results were verified by using the MATLAB platform.

2 New Fractional-Order Chaotic System

It is a special form of the nonlinear chaotic systems; in addition, this form of system has the basic characteristics of an integer order chaotic system, and it also has other characteristics such as extra complexity and comfortable behaviour.

The proposed two quadratic nonlinearity terms are described by dynamics Eq. (1).

$$\frac{d^q x}{dt^q} = z + xy; \; \frac{d^q y}{dt^q} = a - bz - x^2; \; \frac{d^q z}{dt^q} = -x + cy \tag{1}$$

where x, y and z are the variables, a is the system positive parameter, b and c are the system positive (also, they can be zero) parameters, and q is the fractional derivative order. By considering the parameter ($a = 1$), the fixed (equilibrium) points and chaotic attractors are investigated for the suggested system. By solving the following Eq. (2), the equilibrium points (equilibria) of system (1) can be determined.

$$z + xy = 0; \; a - bz - x^2 = 0; \; -x + cy = 0 \tag{2}$$

This results in the equilibrium points of system (1) as.

$$x = \pm\sqrt{\frac{-a}{((b/a)-1)}}; \ y = \frac{1}{c}x; \ z = -\frac{1}{c}x^2 \tag{3}$$

It clear that there are two cases for the solutions of Eq. (2).

Case1: Hidden attractors: When $(b/c) \geq 1$, that gives two states in the solution of Eq. (3), the first state, if $b > c$, yields the solution of Eq. (3) as $x^2 = -$Real, therefore, there are no equilibrium points in this state. The second state, $b = c$ or/and $c = 0$, leads to contradiction in the solution of Eq. (3); i.e., Eq. (3) has no solution. It is easy to demonstrate that the system (1) has no equilibrium points in such cases. Therefore, in this case, hidden attractors are excited. The corresponding hidden attractor projections are shown in Fig. 1.

Case2: Self-excited attractor: When $(b/c) < 1$, the fractional chaotic system (3) has two equilibrium points.

$$E_{1,2} = \left(\pm\sqrt{\frac{-a}{((b/a)-1)}}, \pm\frac{1}{c}\sqrt{\frac{-a}{((b/a)-1)}}, -\frac{1}{c}\left(\frac{-a}{((b/a)-1)}\right)\right)$$

The corresponding Jacobian matrix of the fixed points is obtained as follows.

$$J_{E_{1,2}} = \begin{bmatrix} \pm\frac{1}{c}\sqrt{\frac{-a}{((b/a)-1)}}, & \pm\sqrt{\frac{-a}{((b/a)-1)}} & 1 \\ -2\left(\pm\sqrt{\frac{-a}{((b/a)-1)}}\right) & 0 & -b \\ -1 & c & 0 \end{bmatrix} \tag{4}$$

For determining the stability of the fixed points, the eigenvalues for the Jacobian matrix are calculated according to the following equation.

$$|\lambda I - J_E| = 0 \tag{5}$$

We select the parameters as $a = 1, b = 0.05$ and $c = 0.1$, then the equilibrium points are obtained as $E_{1,2} = (\pm\sqrt{2}, \pm10\sqrt{2}, -20)$ and the corresponding eigenvalues as $\lambda_{1,2,3} = (13.7781, 0.3903, -0.0263)$. That means these fixed points are unstable focus. With fractional order $(q = 0.98)$ and initial condition as $(x_0, y_0, z_0) = (0.1, 0.1, 0.1)$, the corresponding self-excited attractor projections are shown in Fig. 2.

3 Synchronization Strategy

In this section, the synchronization for two identical new fractional-order chaotic systems is considered. A synchronization mechanism has been investigated, so
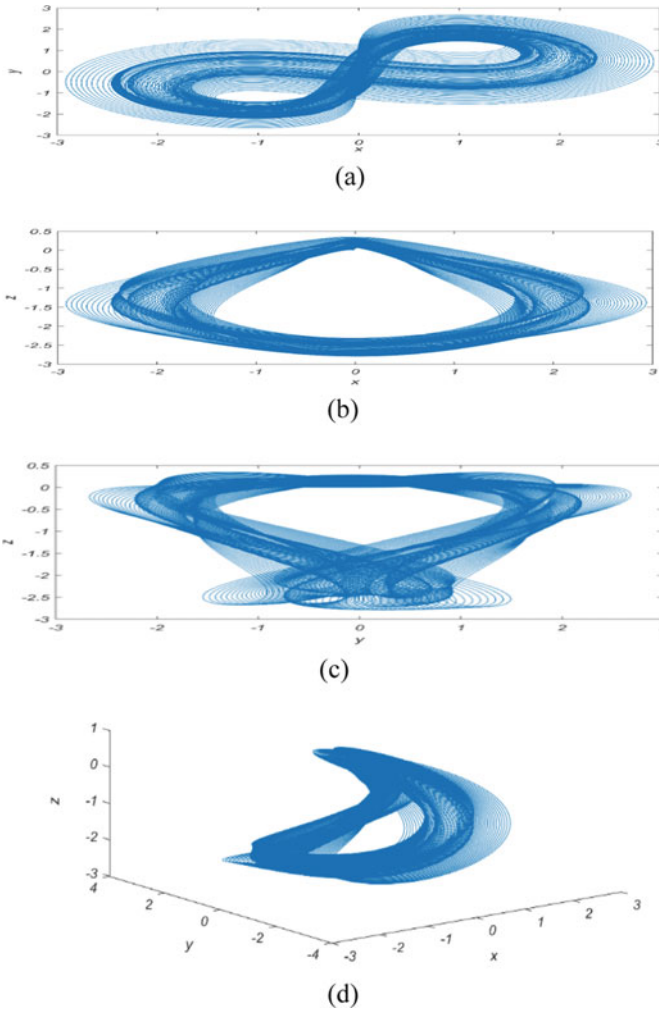
Fig. 1 Hidden chaotic attractor phase portraits of system (1) **a** x–y plane, **b** x–z plane, **c** y–z plane and **d** x–y–z space

that the slave (response system) trajectories asymptotically match the master (drive system) trajectories and verify synchronization. In secure communication applications, chaos synchronization is widely exploited. At the transmitter side, it is a very secure method to hide information in a chaotic signal resulting in the transmission of a secure encrypted signal, that is because chaos exhibits special features as time-based complexity and extreme initial condition sensitivity and unpredictable chaos behaviour. On the other hand, the original information signal can be retrieved by using the chaos synchronization technique at the receiver end. With the growth of research, many control and synchronization mechanisms have been proposed in the
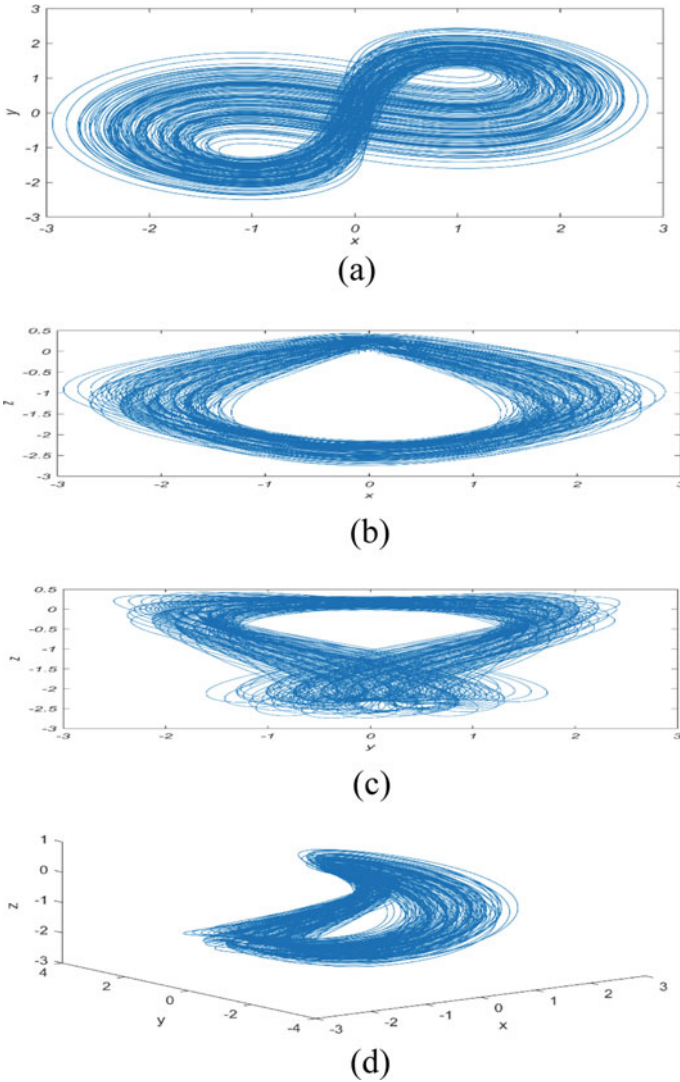
Fig. 2 Self-excited portraits of system (1); **a** x–y plane, **b** x–z plane, **c** y–z plane and **d** x–y–z space

chaos field and its applications especially in secure communication schemes. That is responsible for achieving the synchronization and reducing the synchronization error as much as possible to be zero with very less time [14–20].

3.1 Synchronization Controller Design

In this part, two identical new systems based on the Lyapunov method have been developed, considering the consistent slave–master systems are stated as in Eqs. (10) and (11), respectively.

$$\frac{d^q x_m}{dt^q} = z_m + x_m y_m; \quad \frac{d^q y_m}{dt^q} = 1 - b_m z_m - x_m{}^2; \quad \frac{d^q z_m}{dt^q} = -x_m + c_m y_m \tag{6}$$

$$\frac{d^q x_s}{dt^q} = z_s + x_s y_s + u_1; \quad \frac{d^q y_s}{dt^q} = 1 - b_s(t)z_s - x_s{}^2 + u_2; \quad \frac{d^q z_s}{dt^q} = -x_s + c_s(t)y_s + u_3 \tag{7}$$

In Eq. (11), u_1, u_2 and u_3 present synchronization controllers that need to be designed. The master–slave synchronization errors are defined as.

$$e_x = x_s - x_m; \quad e_y = y_s - y_m; \quad e_z = z_s - z_m \tag{8}$$

Therefore, the dynamic errors are determined as in Eqs. (9)–(11).

$$\frac{d^q e_x}{dt^q} = e_z + y_s z_s - y_m z_m + u_1 \tag{9}$$

$$\frac{d^q e_y}{dt^q} = -b_s(t)e_z - e_b z_m - x_s{}^2 + x_m{}^2 + u_2 \tag{10}$$

$$\frac{d^q e_z}{dt^q} = -e_x + c_s(t)e_y + e_c y_m + u_3 \tag{11}$$

where e_b and e_c are the master–slave parameter estimation errors as in Eqs. (12) and (13), respectively.

$$e_b = b_s(t) - b_m \tag{12}$$

$$e_c = c_s(t) - c_m \tag{13}$$

Therefore, the parameter estimation error can be calculated by.

$$\dot{e}_b = \dot{b}_s(t); \quad \dot{e}_c = \dot{c}_s(t) \tag{14}$$

Selecting the quadratic positive definite Lyapunov function as that results.

$$V\left(e_x, e_y, e_z, e_b, e_c\right) = \frac{1}{2}(e_x^2 + e_y^2 + e_z^2 + e_b^2 + e_c^2) \tag{15}$$

$$\dot{V} = (e_x \frac{d^q e_x}{dt^q} + e_y \frac{d^q e_y}{dt^q} + e_z \frac{d^q e_z}{dt^q} + e_b \dot{e}_b + e_c \dot{e}_c) \tag{16}$$

4 Numerical Simulation Results

These numerical simulations are performed by MATLAB platform. The obtained synchronized states are shown in Fig. 3. Figure 4 confirms that the synchronization errors e_x, e_y and e_z converge to zero quickly with time (in less than 0.3 s). From Fig. 5, uncertain slave parameter $b_s(t)$ and $c_s(t)$, are estimated correctly corresponding to the master parameters $b_m = 0.4$ and $c_m = 0.4$, respectively. The estimation process is verified rapidly with time.

Fig. 3 State variables synchronization of systems
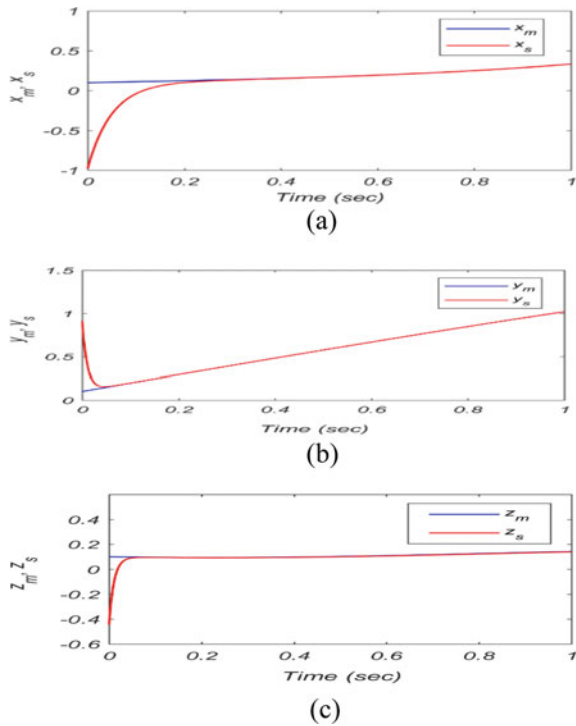
Fig. 4 Synchronization errors evolution e_x, e_y and e_z
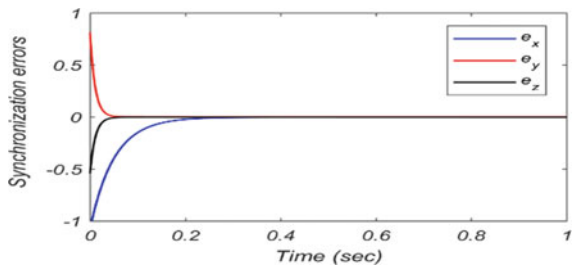
Fig. 5 Slave parameters estimation b_s and c_s
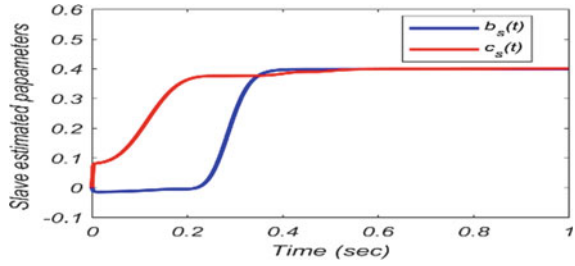
5 Application in Secure Communication

It is possible that fractional-order chaotic systems provide more efficient secure communication when compared with the traditional chaotic systems. It has potential applications in secure communication field and control processing. In Fig. 6, the main two parts of the used secure communication system are the transmitter and the receiver.

Secure communication based on chaos has been applied by many techniques such as chaotic masking, parameter modulation, inclusion and chaotic shift keying. Chaotic masking is the most widely used technique and has been used in this paper. In the chaotic masking technique, the information signal is added to the state output of the chaotic generator at the transmitter resulting an encrypted signal that well be fed to the receiver. The information signal should be weaker than the chaotic output signal by about 20–30 dB which is for hiding the information signal and reaching the synchronization at the receiver. The information signal $m(t)$ has been added to the output chaotic state signal (y_m) at the transmitter side, as follows in Eq. (17).

$$s(t) = m(t) + y_m(t) \tag{17}$$
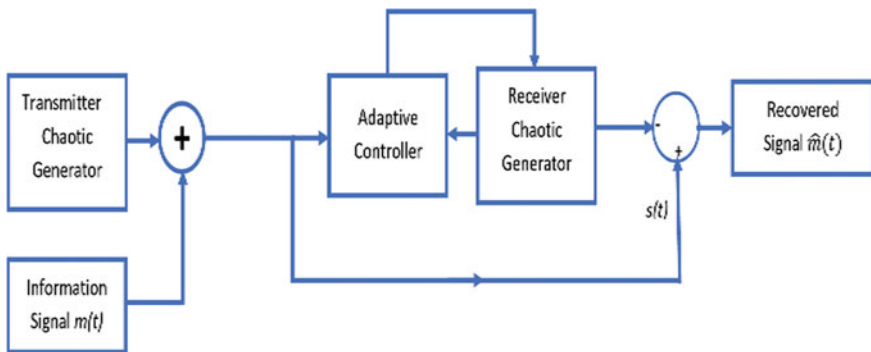
Fig. 6 Block diagram of communication system

where $s(t)$ is the obtained masked signal that well be supplied to the corresponding receiver. Thus, the original information Signal $\widehat{m}(t)$ can be retrieved by subtracting the output chaotic state signal (y_s) from the transmitted signal as in Eq. (18). In the numerical simulation, the message signal is by Eq. (18):

$$m(t) = A\sin(wt) \tag{18}$$

where A is the amplitude and w is the radian frequency of the message sinusoidal wave signal. The obtained numerical simulation results of the chaotic masking for secure communication are displayed in Fig. 7, where the parameters are selected as $a_m = a_s = 1, b_m = 0.4, c_m = 0.4, b_s$ and c_s are estimated corresponding to the matched bm and cm, respectively, $q = 0.98$, and the initial conditions are (0.1,0.1,0.1) and (1,0,−0.5) for the master and slave, respectively. The retrieved original message $\widehat{m}(t)$ can be obtained according to Eq. (19).

$$\widehat{m}(t) = s(t) - y_s(t) \tag{19}$$

Fig. 7 Secure communication signals; **a** The original signal and recovered signal with $A = 0.05$ and $w = 100$ rad/s, **b** The original signal and recovered signal with $A = 0.1$ and $w = 200$ rad/s and **c** the error difference between the original and retrieved signal
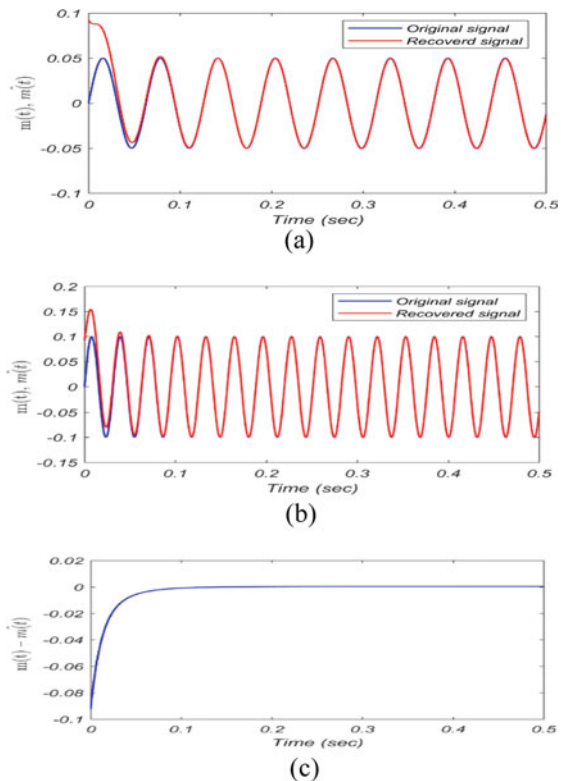
6 Conclusions

The behaviour of dynamics of this proposed system containing the chaotic attractors, the fixed points and corresponding eigenvalues, the bifurcation diagrams, and Lyapunov exponent has been considered. Following that, an adaptive synchronization approach was developed. This synchronization technique is set up in the centre of two indistinguishable new frameworks, one of which fills in as the expert and the other as the slave. The versatile control laws which are for confirming the synchronization have been inferred dependent on Lyapunov steadiness approach. Correspondingly, the unsure slave update boundaries are not really settled for assessing these dubious boundaries. Furthermore, in tumult application field, a solid correspondence plot was built dependent on the created synchronization method. Based on the obtained simulation results, the proposed system's dynamics behaviour is extremely sensitive to initial conditions and small changes in fractional order. Therefore, it has been determined that the proposed new system is effective to be used in high secure communications.

References

1. Zhang S, Zeng YC, Jun Li Z (2018) A novel four-dimensional no-equilibrium hyper-chaotic system with grid multiring hyper-chaotic hidden attractors. J Comput Nonlinear Dyn 13
2. Bi H, Qi G, Hu J, Faradja P, Chen G (2020) Hidden and transient chaotic attractors in the attitude system of quadrotor unmanned aerial vehicle. Chaos Solit Fract 138:109815
3. Lin L, Wang Q, He B, Chen Y, Peng X, Mei R (2021) Adaptive predefined-time synchronization of two different fractional-order chaotic systems with time-delay. IEEE Access 9:31908–31920
4. Kar SK (2010) Optimal control of a linear distributed parameter system via shifted Legendre polynomials. Int J Electr Comput Eng (WASET) 5(5):292–297
5. Ma C, Mou J, Liu J, Yang F, Yan H, Zhao X (2020) Coexistence of multiple attractors for an incommensurate fractional-order chaotic system. Eur Phys J Plus 135:1–21
6. Gupta S, Varshney P, Srivastava S (2021) Whale optimization-based synchronization and control of two identical fractional order financial chaotic systems. J Intell Fuzzy Syst 1–14
7. Tepljakov A (2017) Fractional-order modelling and control of dynamic systems. Springer
8. Vaseghi B, Pourmina MA, Mobayen S (2017) Secure communication in wireless sensor networks based on chaos synchronization using adaptive sliding mode control. Nonlinear Dyn 89:1689–1704
9. Chen X, Park JH, Cao J, Qiu J (2018) Adaptive synchronization of multiple uncertain coupled chaotic systems via sliding mode control. Neurocomputing 273:9–21
10. Azar AT, Vaidyanathan S, Ouannas A (2017) Fractional order control and synchronization of chaotic systems, vol 688. Springer, Berlin
11. Tlelo-Cuautle E, Pano-Azucena AD, Guillén-Fernández O, Silva-Juárez A (2020) Analog/digital implementation of fractional order chaotic circuits and applications. Springer, Berlin
12. Swain S, Biswal K, Tripathy MC, Kar SK (2020) Performance analysis of fractional order sallen-key high-pass filter using fractional capacitors. In: 2020 ınternational conference on computational ıntelligence for smart power system and sustainable energy (CISPSSE), pp 1–5. IEEE

13. Martínez-Guerra R, Gómez-Cortés GC, Pérez-Pinacho CA (2015) Synchronization of integral and fractional order chaotic systems. In: A Differ. Algebr. Differ. Geom. approach with Sel. Appl. real-time, Springer, Berlin
14. Dutta M, Roy BK (2020) A new fractional-order system displaying coexisting multiwing attractors; its synchronisation and circuit simulation. Chaos Solit Fract 130:109414
15. Vaidyanathan S, Volos C (2016) Advances and applications in chaotic systems, vol 636. Springer
16. Subramani C, Dash SS, Kumar V, Kiran H (2012) Implementation of line stability index for contingency analysis and screening in power systems. J Comput Sci 8(4):585
17. Tripathy S, Debnath MK, Kar SK (2021) Jaya algorithm tuned FO-PID controller with first order filter for optimum frequency control. In: 1st Odisha ınternational conference on electrical power engineering, communication and computing technology, ODICON 2021
18. Padhi JR, Debnath MK, Kar SK (2021) LFC investigation with SHO tuned 1 plus fraction ID controller. 1st Odisha ınternational conference on electrical power engineering, communication and computing technology, ODICON 2021
19. Nayak S, Kar SK, Dash SS (2021) Combined fuzzy PID regulator for frequency regulation of smart grid and conventional power systems. Indonesian J Electr Eng Comput Sci 24(1):12–21
20. Biswal K, Swain S, Tripathy MC, Kar SK (2021) Modeling and performance ımprovement of fractional-order band-pass filter using fractional elements. IETE J Res

A Comparative Analysis of Weekly Sales Forecasting Using Regression Techniques

Gopal Behera, Ashutosh Bhoi, and Ashok Kumar Bhoi

Abstract In today's competitive market, every retail store needs to run its business successfully. Therefore they need to know the future demand of customers ahead. So that the retail store can have sufficient stock of their demanding product in their inventory. However, estimating future sales demand is an important and challenging task for any retailer store. In this paper, we use regression techniques to tackle such type of challenging task. Further, we investigate, evaluate, explore the merits and demerits of different regression-based forecasting techniques to predict weekly sales of a store. These forecasting techniques include linear, lasso, ridge, random forest, decision tree, extra tree, and XGBoost regression. Moreover, the effectiveness of the above models is measured through the performance metrics such as MAE, Weighted MAE (WMAE), and RMSE on a publicly available dataset. The experimental outcomes conclude that the random forest technique achieves better results than other methods.

1 Introduction

Forecasting future demand for sales is one of the key aspects for a retailer to make their business run and grow as well as to maintain a successful business in the long run [23]. Whereas Ostrow in 2013 [16] shows that the revenue of companies hikes

G. Behera (✉) · A. K. Bhoi
Department of Computer Science and Engineering, Government College of Engineering Kalahandi, Bhawanipatna, Odisha, India
e-mail: gbehera@gcekbpatna.ac.in

A. K. Bhoi
e-mail: akb@gcekbpatna.ac.in

A. Bhoi
International Institute of Information Technology, Bhubaneswar, India

by 10% of the total revenue of a company over a year due to accurate forecasting, Mindykim in 2019 [15] shows that a company's operation either shrinks or increases if there is no sales forecasting technique is used by a company. Nowadays, social platform plays a significant role in the decision-making process [25]. This allows the retail industry to incorporate advanced technologies in its business. So that it can make an accurate prediction of sales. In this way, corporations can make effective decisions and maximize their resources. For example, Walmart uses machine learning models for sales prediction by analyzing historical data. This also reduces uncertainty and anticipates market changes while influencing investors to act. Over a few decades, sales forecasting with time series plays a vital role not only in academia but also in industry. Therefore many researchers have focused their research in this direction and have developed efficient models to enhance the accuracy of the forecasting model [1]. However, researchers sometimes used time series forecasting techniques without understanding the demerits and applicability of each technique [27]. In fact, using inaccurate prediction techniques in a time series leads to a bad forecast. Hence researchers or forecasters have to select carefully a suitable prediction technique at the time of using the time series model [19].

Predictive and data analytic models are generally used to forecast future sales in sales forecasting. Various classifiers are deployed to forecast sales, such as decision trees, regression, random forest, K-nearest neighbor, cluster analysis, etc. The existing literature is unclear about the different regression models and cannot provide which model can provide better results among all. In this paper, we investigate different sales forecasting techniques and help retailers by testing the performance of the various forecasting techniques based on regression. These techniques that are considered in this article are elaborated in Sect. 3 are linear regression (LR), KNN, lasso, ridge, random forest (RF), decision tree (DT), extra tree, and XGBoost regression. In fact, all these techniques have the capability to handle the trend and seasonality present in the dataset. The main contributions of this work are outlined as:

- We first analyze the different regression techniques with their pros and cons.
- We have implemented eight regression techniques and made comparisons among them.
- We have evaluated the above techniques on the real-world dataset and chose the best technique among all.

The remaining part of the article is structured as: Related work in Sect. 2, and various regression-based forecast techniques are explained in Sect. 3, respectively. Section 4 elaborates the experimental setup with result analysis. Conclusion with the future scope is presented in Sect. 5.

2 Related Work

Over a few decades, ARIMA and HW techniques are commonly used as time series forecasting techniques. All these techniques mentioned above can determine the complex relationship among time series data. However, one stream of literature makes a comparison among five methods such as HW [4], SARIMA, and KNN regression techniques [13] for time series forecasting. Comparisons between ARIMA and state-space models have been discussed in the paper [21] for forecasting women's footwear and found that there is no such difference exists in results among these models. Au et al. [2] have discussed evolutionary neural networks in the fashion retailing domain and found that the model produces more accurate predictions than fully connected networks, also compares the performance evolutionary neural network with SARIMA. A hybrid model has been proposed by Pan et al. [17] that integrates empirical decomposition with a neural network (EMD-NN) to predict sales for a retailer and claimed that EMD-NN produces better performance over classical ANN and SARIMA model when economic conditions are volatile. Refrigerator sales have been analyzed [9] from a retail store. They found that the model has an advantage when modeling input data uncertainty over previous studies. Kolassa [14] used predictive distributions in discrete format for forecasting daily sales of a store and shows why predictive measures are not suitable for count data. Jiménez et al. [12] implemented a feature selection technique to achieve a better prediction for e-commercial sales also found out that similar features of the sold products may affect the sales. Huber and Heiner [11] have discussed the daily demand for product categories of retailer stores using machine learning. The authors do not consider the new product demands and feature importance. Machine learning model such as the XGBoost technique is used for forecasting the future sales of an outlet like Walmart in [24]. The authors have only compared the XGBoost technique with LR and ridge regression. Ranjitha and Spandan [22] have developed a predictive model for forecasting the future sales of Big Mart using Ridge regression, linear regression, polynomial regression, and XGBoost techniques. Subbareddy et al. [20] have analyzed sales on black Friday using six machine learning algorithms to obtain the optimum sales.

Regression is an analytical method that can find the relationship between dependent and independent variables and is used in various fields such as economics, teaching, financing, etc. Linear regression is a single independent variable model for forecasting the result of the dependent variable. In contrast, multiple regression has two or more independent variables to produce the result of the dependent variable [13]. Pavlyshenko [18] in 2019 has shown that the utilization of regression approaches can give more reliable outcomes as opposed to time series techniques. Hence, sale forecasting is a regression problem rather than a time series problem. To our knowledge, there are no articles that compare more than five models simultaneously. Therefore, in this paper, we analyze all the regression techniques with their merits and demerits and make a comparison among them to forecast the sales of a store.

3   Forecasting Techniques

In this section, we have discussed various forecasting techniques based on regression and explored their pros and cons.

3.1   *Linear Regression (LR)*

Linear regression builds [13] a relationship between the dependent variable (Y) and independent variable (X) using a regression line. The regression line is defined in Eq. 1.

$$Y = A + B^*X + E \tag{1}$$

where A is the intercept, B is the slope of the regression line and E is the error term. For a given predictor LR is to predict a value for a target variable. To obtain a best-fit line we need to minimize the sum of the squares as defined in Eq. 2.

$$\min_{w} \| X_w - y \|_2^2 \tag{2}$$

Advantages

1. Linear regression is simple and easy to understand among forecast techniques.
2. It provides Interpretability, scientific acceptance, and wide spread availability.

Disadvantages

1. The main disadvantage is that many real-world phenomena do not correspond to the assumptions of the linear model as they produce a bad result.

3.2   *K Nearest Neighbor (KNN) Regression*

KNN is a simple forecasting technique [13] which stores all available cases and predict target case using a similarity score (e.g., distance functions). The distance can be computed either by using Euclidean, Manhattan, or hamming distance, etc., whereas Euclidean distance is a frequently used measure and defined in Eq. 3.

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{3}$$

Advantages

1. Does not require training period: KNN is sometimes called lazy learner as it does not learn anything in the training time.

2. New data can be easily added as KNN does not require training before making predictions.

Disadvantages

1. KNN does not handle large datasets: The cost of the calculating the distance between the new point and each given point in the dataset will take more time, which will degrade the performance.
2. Sensitive to outliers, noisy data, and missing values: KNN is very sensitive to noise and outliers if present in the data. Therefore outliers need to be removed.

3.3 Ridge Regression (RR)

Ridge regression [18] overcomes the multicollinearity problem that existed in the least-square estimate and uses l_2 regularization to avoid the overfitting. RR reduces the error by adding bias to produce a better prediction and is defined in Eq. 4 for multiple independent variables.

$$Y = a + b_1 x_1 + b_2 x_2 \cdots + e \qquad (4)$$

This forecasting technique resolves the problem of multicollinearity through shrinkage parameter λ, as mentioned in Eq. 5. Where the equation consists of two components, the first one is the lasso and the second component indicates the penalty term, whereas β represents the regression coefficient.

$$\underset{\beta \in R}{\arg\min} \|Y - X\beta\|_2^2 + \lambda \|\beta\|_2^2 \qquad (5)$$

Advantages

1. Avoids overfitting: use of regularization makes ridge regression to overcome the overfitting problem.
2. Ridge regression performs well in cases of large multivariate data with the number of $p > n$ (predictors > number of observations).

Disadvantages

1. Ridge regression is unable to perform feature selection.
2. There is a trade-off between variance and bias.

3.4 Lasso (Least Absolute Shrinkage and Selection Operator) Regression

Lasso is equivalent [18] to RR but uses l_1 regularization instead of l_2 and adds a penalty equal to the absolute size of the coefficients. It is able to reduce the variability and enhance the accuracy of the LR model. Lasso regression is defined in Eq. 6 which leads to penalizing values of some parameters that became exactly to zero.

$$\underset{\beta \in R}{\operatorname{argmin}} \|y - X\beta\|_2^2 + \lambda \|\beta\|_1 \qquad (6)$$

Pros

1. Lasso improves performance over ridge and OLS by penalized regression.
2. Both variable selection and shrinkage are done simultaneously for better forecasting.

Cons

1. It selects at most n variables before it saturates.
2. The model selected by lasso is not stable. For example, on different bootstrapped data, the feature selected may be completely different.

3.5 Decision Tree Regression

A decision tree (DT) [3] technique is used as regression when the target variable is discrete/continuous. This regression technique uses standard deviation or least-square reduction as a measure to select attributes for the regression tree. The least-square or standard deviation between two variables is defined in Eq. 7.

$$S(\text{Target,Predictor}) = \sum_{c \in \text{Predictor}} P(c)s(c) \qquad (7)$$

where $P(c)$: the probability of attribute value c present in predictor and $s(c)$ is the standard deviation of c.

Advantages

1. DT algorithm requires less effort as compared to other methods for preparing data during preprocessing.
2. A DT does not require normalization and scaling of data.

Disadvantages

1. A small change in the data causes a significant change in the decision tree structure, causing DT instability.

2. Decision tree often takes more time to train the model which makes it relatively expensive and more complex.

3.6 Random Forest Regression

Random forest (RF) [26] is an ensemble of DT. That is many trees are constructed in a random way to form an RF. Thus, the tree is built by selecting a sample of rows and then splitting them into different elements at each node. Every tree makes their own prediction, and average of all prediction is used to obtain a single result; due to this, an RF performs better than a single DT.

Advantages

1. RF can be used to handle missing values in the given data also solves the overfitting problem.
2. High dimensional data is handled very well in RF.

Disadvantages

1. It is computationally expensive and very difficult to interpret.
2. A large number of trees take ample time which, makes predictions quite slow.

3.7 Extra Tree Regression

The extra tree technique (extremely randomized trees) [10] is an ensemble of DT, with further randomizing tree building, where each tree is trained using the whole learning samples instead of bootstrap sample and it selects a cut point randomly rather than finding an optimal cut point for each K features, chosen randomly at each node.

Pros

1. Extra tree is worse in case of high dimensional data.
2. Extra tree provides better performance in case of data containing noisy features.

Cons

1. When all the variables are relevant, both methods seem to achieve the same performance.

3.8 XGBoost Regression

XGBoost is an optimized [5, 7] distributed gradient boosting tree to enhance the accuracy of trees and is a highly efficient, flexible, and portable gradient boosted tree

(GBT). XGBoost is based on function approximation by optimizing loss function as shown in Eq. 8. Due to parallelism tree boosting mechanisms like Gradient boosting mechanism (GBM) and gradient boosting decision tree (GBDT), this technique resolves many problems of data science in a fast and accurate way.

$$L^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \omega(f_t) \tag{8}$$

where ω: regularization term used to avoid overfitting, y_i : actual forecasting, $\hat{y}_i^{(t-1)}$ is the predicted value and f_t represents the set base learner that is $F = \{f_1, f_2, \cdots, f_m\}$. So that the final forecasting is represented as $\hat{y}_i = \sum_{t=1}^{m} f_t(x_i)$.

Pros

1. Extremely fast (parallel computation) and highly efficient.
2. Versatile: That is XGBoost can be used for purpose of regression, classification, or ranking and does not require feature engineering.

Cons

1. The biggest disadvantage is that XGBoost works only on numeric features.
2. If hyper-parameters are tuned properly then it leads to overfitting.

4 Implementation

We have set up an experiment in Jupyter Notebook environment with python and used a publicly available dataset for testing the performance of the various models. Where a brief description of the dataset is discussed in Sect. 4.1.

4.1 Dataset

In this article, we have used the publicly available Walmart[1] dataset, which contains 45 stores around the globe. Each store consists of a number of departments as shown in Table 1. In addition, Walmart runs several promotional offers throughout the year. These offers include prominent holidays, thanksgiving, labor day, the four largest of which are the Super Bowl, and Christmas.

---

[1] https://www.kaggle.com/c/walmart-recruiting-store-sales-forecasting/data.

Table 1 Details of Walmart store dataset

Name of the attribute	Description
Store	Total 45 store
Type	Walmart contains three type of store type such as A, B, and C
Size	Number of product contained in a particular store(34,000–210,000)
Fuel Price	Price of fuel in the particular area
Mark Down 1:5	Indicates offer type or markdown
CPI	Weekly Consumer Price Index
Unemployment	Unemployment rate in a particular week
Date	Observation date
Weekly_Sales	Sales record per week
Dept	There are 99 departments
IsHoliday	Whether week is a Holiday week or not

4.2 Results and Discussion

To measure the performance of the forecasting model we consider the following performance measure metrics, namely RMSE (root mean square error) [5], MAE (mean absolute error) [6], and WMAE (weighted MAE) [8]. Smaller the RMSE, MAE, and WMAE, indicates better forecasting. RMSE is defined in Eq. 9.

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum\nolimits_{i=1}^{N} (f_i - \hat{f}_i)^2} \tag{9}$$

where f_i is the actual value, $\hat{f}_i$ is the predicted value and N represents a number of samples. Similarly, Eqs. 10 and 11 denote MAE and WMAE respectively.

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^{N} |f_i - \hat{f}_i| \tag{10}$$

$$\text{WMAE} = \frac{1}{\sum w_i} \sum_{i=1}^{N} w_i |f_i - \hat{f}_i| \tag{11}$$

where w_i : are the weights and $w_i = 5$ if the week is holiday otherwise the $w_i = 1$. Before testing the model performance, we first analyze the dataset with the help of Exploratory Data Analysis to see the behavior of the dataset. Figure 1a shows that highest and lowest sales are obtained at store number 10 and 44 respectively. Further,
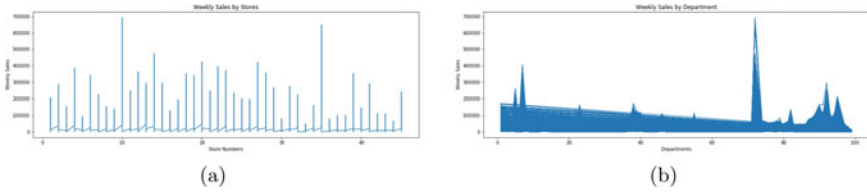
Fig. 1 Weekly sales with respect to store and department of Walmart
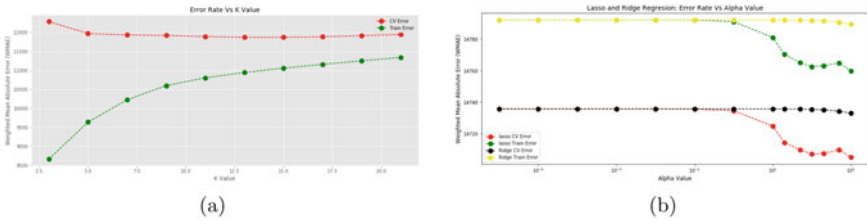
Fig. 2 Error rate with respect to K and α in KNN and Lasso regression
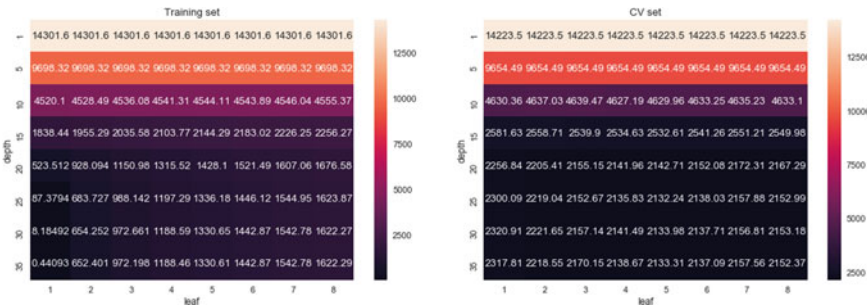
Fig. 3 Error rate vs maximum_depth and number of leaf on decision tree regression

from Fig. 1b, it is found that sales record of maximum departments is below 200,000 unit.

For evaluation purposes, the dataset is split randomly with training–testing split with 80:20 ratio, respectively. Then all models are trained on the training set and validated on testing set to find out the training loss and validation loss of every model. The error rate with respect to k in KNN regression is found in Fig. 2a, which indicates that both training and validation errors are low at $k = 11$. Similarly, Fig. 2b represents error rate w.r.t α in lasso and ridge regression and found that both training and validation errors are low at $\alpha = 1000$.

Whereas Fig. 3 indicates that the errors with respect to maximum depth and number of leaves in decision tree regression and found that both errors are least for maximum_depth $= 25$ and minimum sample leaf $=5$.

Further both training and validation error of random forest, extra tree and XGBoost regression are shown in Figs. 4, 5 and 6 respectively. Whereas for random forest and
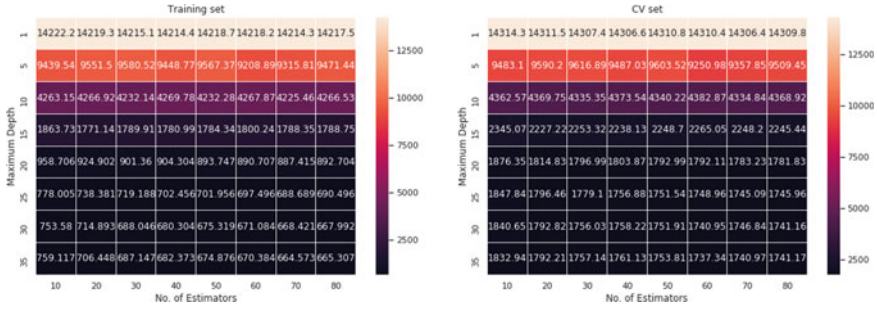
Fig. 4 Error rate vs maximum_depth and number of estimator on random forest regression
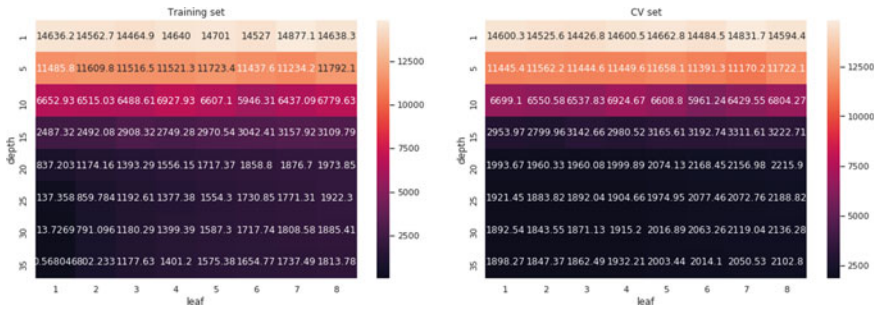
Fig. 5 Error rate versus maximum_depth and number of leaf on extra tree regression
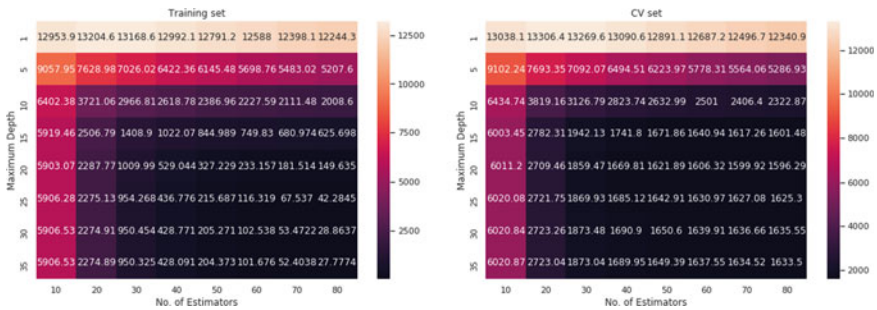
Fig. 6 Error rate versus maximum_depth and number of estimator on XGBoost regression

XGBoost regression both errors are low for maximum_depth $= 15$ and number of estimators equal to 20 in XGBoost and for random forest the number estimator is 80 and depth is 35. In the case of the decision tree, both errors are low at $depth = 25$ and the number of samples leaf is 5. Similarly, for extra tree regression depth is 30 and the minimum samples leaf is 2.

At last, the performance comparison of eight regression models are shown in Table 2 and found that the random forest technique is the best forecasting technique among the models.

Table 2 Comparison of performance of Different forecasting techniques.

Forecasting techniques	RMSE	MAE	WMAE
Linear Regression	21772	14614.16	14834.9
KNN Regression	18853.47	11565.47	11887.66
Ridge Regression	21772.41	14613.28	14831.56
Lasso Regression	21774.52	14608.3	14797.6
Decision Tree Regression	4779.21	1890.89	2123.09
Random Forest Regression	**3870.91**	**1555.07**	**1754.91**
Extra Tree Regression	4132.88	1619.74	1902.56
XGBoost Regression	5750.72	2632.39	2778.17

5   Conclusion and Future Work

This paper analyzes several regression techniques to tackle the forecasting challenges. Further, this article presents the weakness and strengths of each technique. For the evaluation purpose, first, we conduct exploratory data analysis and conclude that the weekly sales are the highest for store number 14 and low at store number 44. Similarly, weekly sales are influenced by the department. That is most of the departments have sales below 200,000 units.

Further to know the effectiveness of the models, we have conducted several experiments and analyzed training and cross-validation errors. It is worth noting that the training and validation errors are low at $k = 11$ for KNN, whereas for Lasso, Ridge regression, the errors are low at $\alpha = 1000$. In the case of tree-based regression, the errors are low, when the estimator $= 20$ and maximum_depth $= 15$ for XGBoost and $estimator = 80$ and maximum_depth $= 35$ for the random forest, respectively. Similarly, in the decision tree, the error is low at depth $= 30$ and estimator $= 25$. Moreover, we conclude from the performance metrics that the random forest technique is the best forecasting technique among all. In the future, we will develop an optimized forecasting technique and a deep learning technique to enhance forecasting performance in this domain.

References

1. Adhikari R, Agrawal RK (2013) An introductory study on time series modeling and forecasting. arXiv preprint arXiv:1302.6613
2. Au KF, Choi TM, Yu Y (2008) Fashion retail forecasting by evolutionary neural networks. Int J Prod Econ 114(2):615–630
3. Behera G (2011) Privacy preserving c4. 5 using gini index. In: 2011 2nd National conference on emerging trends and applications in computer science. IEEE, pp 1–4

4. Behera G, Bhoi AK, Bhoi A (2020) Uhwsf: univariate holt winter's based store sales forecasting. In: Intelligent systems: proceedings of ICMIB 2020, p 283
5. Behera G, Nain N (2019) A comparative study of big mart sales prediction. In: CVIP, vol 1, pp 421–432
6. Behera G, Nain N (2019) Grid search optimization (GSO) based future sales prediction for big mart. In: 2019 15th International conference on signal-image technology & internet-based systems (SITIS). IEEE, pp 172–178
7. Chen T, Guestrin C (2016) Xgboost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pp 785–794
8. Cleger-Tamayo S, Fernández-Luna JM, Huete JF (2012) On the use of weighted mean absolute error in recommender systems. In: RUE@ RecSys. pp 24–26
9. Fabianová J, Kačmáry P, Molnár V, Michalik P (2016) Using a software tool in forecasting: a case study of sales forecasting taking into account data uncertainty. Open Eng 1(open-issue)
10. Geurts P, Ernst D, Wehenkel L (2006) Extremely randomized trees. Mach Learn 63(1):3–42
11. Huber J, Stuckenschmidt H (2020) Daily retail demand forecasting using machine learning with emphasis on calendric special days. Int J Forecast 36(4):1420–1438
12. Jiménez F, Sánchez G, García JM, Sciavicco G, Miralles L (2017) Multi-objective evolutionary feature selection for online sales forecasting. Neurocomputing 234:75–92
13. Kohli S, Godwin GT, Urolagin S (2020) Sales prediction using linear and KNN regression. In: Advances in machine learning and computational intelligence. Springer, pp 321–329
14. Kolassa S (2016) Evaluating predictive count data distributions in retail sales forecasting. Int J Forecast 32(3):788–803
15. mindykim: the 5 financial impacts of missing your next sales forecast—datahug, 2019. https://datahug.com/blog/5-financial-impacts-missing-next-sales-forecast/ (2019)
16. Ostrow P (2013) Sales forecasting, how top performers leverage the past, visualize the present, and improve their future revenue. Recuperado el
17. Pan Y, Pohlen T, Manago S (2013) Hybrid neural network model in forecasting aggregate us retail sales. In: Advances in business and management forecasting. Emerald Group Publishing Limited
18. Pavlyshenko BM (2019) Machine-learning models for sales time series forecasting. Data 4(1):15
19. Puthran D, Shivaprasad H, Kumar KK, Manjunath M (2014) Comparing SARIMA and holt-winters' forecasting accuracy with respect to Indian motorcycle industry. Trans Eng Sci 2(5):25–28
20. Ramasubbareddy S, Srinivas T, Govinda K, Swetha E (2021) Sales analysis on back friday using machine learning techniques. In: Intelligent system design. Springer, pp 313–319
21. Ramos P, Santos N, Rebelo R (2015) Performance of state space and ARIMA models for consumer retail sales forecasting. Robot Comput Integr manufacturing 34:151–163
22. Ranjitha P, Spandana M (2021) Predictive analysis for big mart sales using machine learning algorithms. In: 2021 5th International conference on intelligent computing and control systems (ICICCS). IEEE, pp 1416–1421
23. Rao SK (1985) An empirical comparison of sales forecasting models. J Prod Innov Manag Int Publ Prod Dev Manag Assoc 2(4):232–242
24. Shilong Z et al (2021) Machine learning model for sales forecasting by using xgboost. In: 2021 IEEE International conference on consumer electronics and computer engineering (ICCECE). IEEE, pp 480–483
25. Syam N, Sharma A (2018) Waiting for a sales renaissance in the fourth industrial revolution: Machine learning and artificial intelligence in sales research and practice. Industrial marketing management 69:135–146
26. Tyralis H, Papacharalampous G (2017) Variable selection in time series forecasting using random forests. Algorithms 10(4):114
27. Winters PR (1960) Forecasting sales by exponentially weighted moving averages. Manag Sci 6(3):324–342

Complex Network Visualisation Using JavaScript: A Review

Sumit Dutta and Swarup Roy

Abstract Real-world networks are very complex in nature. Unlike simple graphs, they follow unconventional topological properties because of which they are treated as *complex networks* or graphs. Availability of an effective computing platform for processing this kind of network is a need these days. Advanced logical inquire about exercises request overwhelming utilise of web-based technology for covered up information disclosure and sharing of graph data. This can be accomplished through the development of extremely interactive web-enabled complex graph visualisation and analysis tools. Such platforms may help in the process, analyse and visualise complex networks on the web. Modern internet technologies such as HTML5, CSS3, Neo4j, and robust JavaScript libraries can also be used effectively to tackle large graph datasets to visualise effectively over the internet. This paper presents an overview of the highlights of a few openly accessible JavaScript libraries for developing interactive web platform for analysing and visualising complex graphs. We also demonstrate how to implement a few important analysis and visualisation features using one of the most popular and latest JavaScript libraries, *D3.js*. We show the use of the R wrapper package, *networkD3*, of D3.js to visualise few sample graph properties.

Keywords Complex graph · Visualisation · D3.js · Networkd3 ·
R programming · Graph analysis tool

1 Introduction

Real-life systems are large, complex, dynamic and ubiquitous. For example, WWW, the internet, wireless networks, supply-chain networks and most predominantly social [10] and biological networks [1] are the few of many. Often the complex inter-relationship between different participating entities of the system are represented mathematically as graph or network [7]. In the context of graph concept, the nodes in a social network is a collection of members or actors, and edges denote the col-

S. Dutta · S. Roy (✉)
Network Reconstruction & Analysis (NetRA) Lab, Department of Computer Applications,
Sikkim University, Gangtok, Sikkim, India
e-mail: sroy01@cus.ac.in

laboration or association among themselves [9]. In case of bio-molecular systems, a complex interaction among macromolecules such as Protein-Protein interaction (PPI), Genetic regulation, Signalling Pathways, Metabolic activities, Neuronal connectivity within a brain etc. are generally modelled as graph or network. Be that as it may, such graphs are not straightforward graphs like normal graphs. As a result of their unconventional topological properties, they are most commonly dealt with as *complex graphs* or networks. Unlike conventional graphs, real-world networks are characterised by non-trivial topological features like degree distributions, clustering coefficients, degree assortativity or average path length of the network [7].

The human brain has advanced momentous visual handling capabilities to examine designs and pictures. Accordingly, an interactive visual illustration of complex information systems is a popular procedure of choice to simplify the interpretation of information. It is challenging yet of utmost importance to develop a suitable computing platform for understanding the interaction structure and connectivity dynamics of any complex network. Complex network visualisation and analysis tools involve collecting quantitative methods and graphical rendering of the network to unveil network dynamics and functionality. Numerous open-source tools are readily available for the same,[1] with their relative merits and demerits. Development of similar tools with added features needs effective visualisation and analysis libraries that can easily be integrated with the new tool to be built. The web platform is the most obvious choice for the developers due to the ubiquitous accessibility of the platform.

To develop highly interactive web applications, web designers usually prefer to utilise the power of JavaScript (JS)[2] library or framework rather than any other options. Due to the open-source nature, JavaScript technologies keep on evolving. For the last decades, JS experienced a significant futuristic advancement towards handling gigantic scientific knowledge sets to analyse, proffer and visualise on the web. The open-source JS libraries have completely revolutionised the system by which we present and visualise data on the web, whether it is scientific or non-scientific data.

We perform a comprehensive study on various complex network analysis and visualisation using JS libraries, available freely. We report their comparative features that may help developers select a suitable library according to their need. We demonstrate using one of the popular JS libraries, D3.JS, to implement a few basic network visualisation features.

2   JavaScript Libraries for Network Visualisation

Here, we discuss a few promising and popular JavaScript libraries that are unreservedly accessible and can be used to visualise and handle large complex networks. A feature-wise comparison among the libraries is also reported. Any of the libraries

---

[1] https://www.kdnuggets.com/2015/06/top-30-social-network-analysis-visualization-tools.html.

[2] https://www.javascript.com.

could be chosen if it is the best fit for the user's needs, which necessitate fascinating interactivity.

D3.js: D3.js [3] is used for data-driven document manipulation [4, 8]. It is intended to fully exploit the capabilities of modern web requisites such as HTML5, Cascading Style Sheets (CSS3), and SVG (Scalable Vector Graphics). D3.js is the visualisation library that replaces Protovis visualisation library [3]. The Stanford Visualization team at Stanford University created and maintains it. D3.js offers effective animations, interactions and complicated & dynamic visualisations on the web. D3.js accentuates the productive control of HTML elements. It employments pre-built JavaScript capacities in its bit to choose components, make SVG objects, they can be designed or transitions and dynamic effects can be added to them. It is also a great library to use visualisation for complex networks for interactive animation in which user can handle large networks without any clumsiness of the network.

Cytoscape.js: An open-source project initiated by NRNB (U.S. National Institutes of Health, National Centre for Research Resources). JavaScript is used to develop Cytoscape.js [6]. A graph library that are used to analyse and visualise graphs. It's a component of the jQuery plugin. Cytoscape Web is the inheritor to Cytoscape.js [6]. Cytoscape.js incorporates all the motions like box choice, coordinated charts,, pinch-to-zoom, panning, blended charts, and bolster for diverse chart hypothesis client cases, compound charts, undirected charts, multigraphs. Cytoscape.js could be a great library, in spite of the fact that new version of it has been released.

Sigma.js [5]: It is yet another open-source JS library developed by Sciences Po Media Lab. Sigma [5] library is a chart drawing library that provides a simple way for clients to display graphs on web pages either using HTML5 Canvas or WebGL (Web Design Library). It allows application builders to combine network visualisation with content wealthy web platform. Sigma is lightweight, is generally simple to utilise, can be effectively coordinated into existing web applications, and gives different built-in highlights, like HTML5 canvas and WebGL renderers. Moreover, different from general graph visualisation tools displaying large graphs on the web page, Sigma.js [5] is effective in dynamics graph generation in real-time. It permits visualising interactive graphs derived from foreign sources such as Gephi [2].

Vis.js: A visualisation library that offers browser integrated dynamic visualisation. The library is set up in such a way that it can be handy to make use of, manage huge amounts of dynamic data and allow processing of the information. Almende B.V, a Dutch research corporation developed and managed the library. The main components of Vis.js are DataSet, DataView, Network, Graph2d, Graph3d, and Timeline. But it fails to handle large networks, which can be a drawback for visualising large complex networks.

Springy.js: Springy offers aesthetically pleasing graph layout. The purpose of Springy.js is to provide simple as well as straightforward graph. It offers to deliver

---

[3] https://d3js.org.

relevant graph manipulation and representation techniques. Moreover it also supports various rendering engines like Canvas, SVG, WebGL, or even HTML elements.

JointJS.js: It's another JS framework for visualisation that works with HTML5 and allows you to interact with diagrams and charts. Static charts, interactive diagrams, and other types of application builders can all be created using it. When it comes to model visualisation with interactivity, JointJS makes a difference.

Arbor.js: Arbor is made with the help of web workers thread and jQuery script. This plugin includes the force-directed version of graph layout, which is fast in nature. In addition to this, it also offers APIs for graph organisation and handling screen refreshes. However, it gives freedom to the user for actual screen-drawing. This implies that the user can integrate Arbor with canvas, SVG, or even positioned HTML elements as per requirement.

We report synopsis of different graph visualisation JavaScript libraries in Table 1.

3 *D3*—A Preferred Large Graph Visualisation Platform

Large graphs need to be visualised dynamically and interactively so that each node can be sighted out and its property can be envisaged. Apart from the other visualisation libraries, D3 is more advantageous to visualise complex graphs. Few key points are highlighted below.

1. D3.js is open-source, hence source code may be customised or reused according to the need.
2. D3 works with popular web technologies like HTML, CSS and SVG. No further technologies are required to work with D3.
3. D3 is flexible and does not offer any specific feature that provide a programmer complete control over visualisation.
4. It is fast and works well with large networks. It is lightweight and works directly with different web standards, .
5. It is also suitable for the R programmers too as good R wrapper libraries like *networkD3* are available for the same, which can render and visualise large complex network more effective way.

Though D3.js suitable to visualise large networks, direct use of D3 JavaScript library is associated with few implementation-related issues. It is some time not convenient to use D3.js directly for effective rendering and interactive visualisation of the large graph.

The *networkD3* [4] is an R package built on as a wrapper with comprehensive library of D3.js that makes it easy to visualise large graphs using R code. The htmlwidgets framework uses d3Network to create D3 network graphs. The htmlwidgets framework immensely simplifies the package's syntax for exporting the graphs. Further, it

---

[4] https://CRAN.R-project.org/package=networkD3.

Table 1 Synopsis of various JavaScript libraries for network visualisation

Name	Description	Support categories	Corresponding R package	Supports JSON	Rendering engine	Supported input file formats
D3.js	Protovis' descendent, a powerful multi-use JavaScript visualisation toolkit[1]	Networks/Charts/ Hierarchies	networkD3	yes	SVG/CANVAS	CSV,TSV,JSON,XML
Cytoscape.js [6]	A graph library that are used to analyse and visualise graphs[2]	Networks	Not available	Yes	SVG	SIF, GML, XGMML, CSV, TSV
Sigma.js	Sigma is a small JavaScript library that focuses on graph sketching. It allows developers to integrate network exploration into rich web applications [3]	Networks	Not available	Yes	CANVAS/WebGL	JSON
Alchemy.js	A graph plotting application written pretty much using d3 library[4]	Networks	Not available	Yes	SVG/CANVAS	TXT, CSV, TSV
Vis.js	A browser-based, dynamic visualisation library. The library is simple, that can handle massive volumes of dynamic data, and also allows data manipulation on-the-fly[5]	Networks/Charts	visNetwork	Yes	CANVAS	JSON
arbor.js	A graph visualisation library using web workers and jQuery script. [6]	Networks	Not available	Yes	CANVAS/SVG	JSON
Springy.js	A force-directed graph layout algorithm[7]	Networks	Not available	Yes	CANVAS/SVG/WebGL	JSON
JointJS.js	A JS library for representing networks and diagrams that can be visualised in an interactive way[8]	Networks/Charts/ Hierarchies	Not available	Yes	SVG	JSON
dagre.js	A JavaScript library that makes laying out directed graphs on the client side[9]	Networks	Not available	Yes	SVG	JSON

1 http://d3js.org
2 http://js.cytoscape.org
3 http://sigmajs.org
4 http://graphalchemist.github.io/Alchemy/
5 http://visjs.org/
6 http://arborjs.org/
7 http://getspringy.com/
8 http://js-graph-it.sourceforge.net/index.html
9 https://github.com/cpettitt/dagre

has an extra feature to integrate the Viewer pane's interface with RStudio. It offers a straightforward way to visualise a large graph without knowing the actual D3 code. The advantages of using networkD3 over D3.JS are listed below.

- **Convenient large input file handling**: Though D3 supports JSON format for convenient storing and transporting data, however, often it has been observed it is expensive to handle large input network in various other formats that needs to be converted in JavaScript. This can be solved using R code which supports data frames. Hence, in networkD3, large input files can efficiently work with the data frame, and user can modify and work conveniently according to their needs.
- **Minimisation of coding**: D3 is a huge library. It is often complicated to handle large network and write complex code for analysing the properties of each node and visualising them. As networkd3 is a wrapper library, the same visualisation can be obtained with minimum codes and no knowledge of the actual D3.js library.
- **Easy integration with graph analysis libraries:** Network analysis is often considered as an integrated activity with visualisation for large network. D3 is a visualisation library not for analysis. Often networks are used to visualise the non-trivial properties or analyse the properties of a network. This helps to examine the relationship among entities of a network. *igraph* [5] is an R library for network analysis that can be integrated with networkD3. It can be used with networkD3 to analyse a network and can be rendered the network in the web using D3 features.

3.1 Visualising Networks Using NetworkD3: A Demonstration

The networkD3 package currently supports four types of networks, i. Force-directed networks, ii. Sankey diagrams, iii. Radial networks, and iv. Dendro networks. Below we provide sample R code and their outcomes for three of the networks.

SimpleNetwork: Creates very basic simple D3 JavaScript force-directed network graphs.

```
# Load package
library(networkD3)

# Create data
src <- c("a", "a", "a", "a", "b", "b", "c", "c", "d")
target <- c("b", "c", "d", "j", "e", "f", "g", "h", "i")
networkData <- data.frame(src, target)

simpleNetwork(networkData) # Visualizing the graph
```

[5] https://igraph.org/.

ForceNetwork ForceNetwork gives more control over the appearance of force-directed network, as well as the ability to plot more sophisticated networks.

```
# Load data
data(Links)
data(Nodes)

# Visualizing the graph
forceNetwork(Links = Links, Nodes = Nodes, Source = "source", Target =
"target", Value = "value", NodeID = "name", Group = "group", opacity =
0.8)
```

SankeyNetwork The use of Sankey diagram also called as flow diagram shows the width of the arrows is proportional to the flow rate.

```
# Visualizing the graph
sankeyNetwork(Links = links, Nodes = nodes, Source = "source",
Target = "target", Value = "value", NodeID = "name", units = "TWh",
fontSize = 12, nodeWidth = 30)
```

Graph analysis with igraph The pseudocode for betweenness centrality of a network is shown below:

```
library(igraph)
library(htmlwidgets)
library(networkD3)

data_1 = read.table(file = read_file_path/, sep = '\t', header=
TRUE)

bet <- betweenness(net)
data <- igraph_to_networkD3(net)
bet <- as.numeric(bet)

data$nodes$value <- bet
network <- forceNetwork(Links = data$links, Nodes = data$nodes,
Source = "source", fontSize = 100,Target = "target", NodeID =
"value", Group = 'name', Nodesize ="value", opacity = 0.8,
linkColour = "#000", radiusCalculation =
JS("Math.sqrt(d.nodesize)+6"))

saveNetwork(network,file = 'betweenesnetwork.html')
```

We rendered sample network with few annotated centrality and topological analysis outcomes and presented in the Fig. 1.

(a) Variable size nodes according to betweeness centrality score

(b) Shortest path from one node to the other

(c) Highlighted Page rank centrality value of a node

(d) Variable size node according to node degree
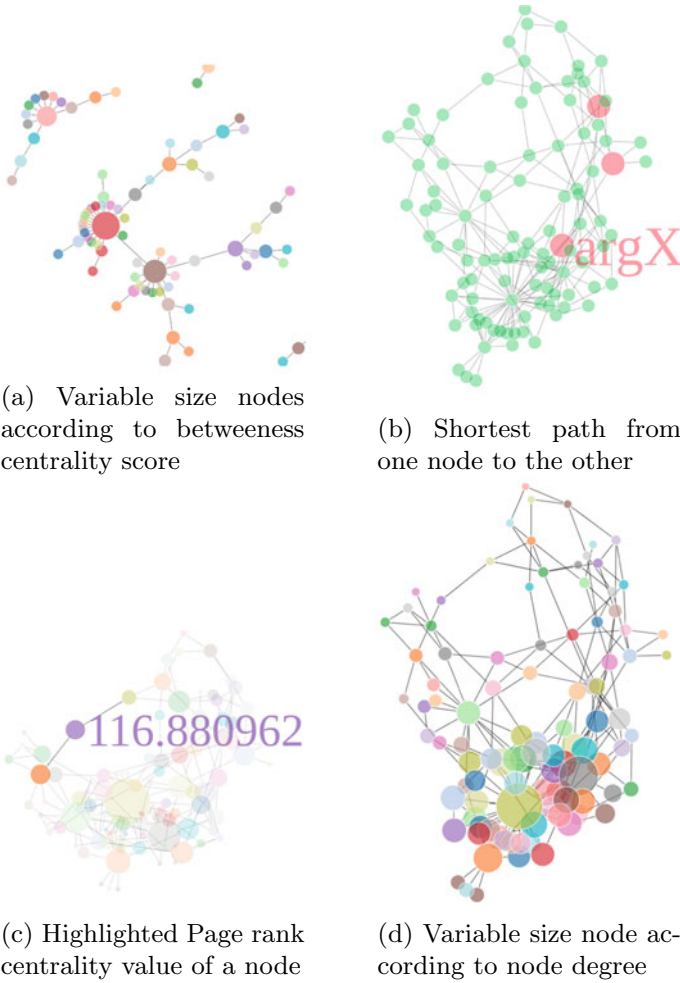
Fig. 1 Visualisation of simple graph with topological features

4   Conclusion

We presented a comprehensive study on the various JavaScript libraries for visualising large networks. We highlighted D3.js as one of the most widely used visualisation libraries. Cytoscape.js is another enriched JS library for complex network visualisation in life sciences. D3.js is the only JavaScript library available in R. networkD3 is a D3.js based R library that can be used to visualise and analyse large graphs efficiently and effectively.

References

1. Ahmed SS, Roy S, Kalita J (2018) Assessing the effectiveness of causality inference methods for gene regulatory networks. IEEE/ACM Trans Comput Biol Bioinf 17(1):56–70
2. Bastian M, Heymann S, Jacomy M (2009) Gephi: an open source software for exploring and manipulating networks. In: Third international AAAI conference on weblogs and social media
3. Bostock M, Heer J (2009) Protovis: a graphical toolkit for visualization. IEEE Trans Visua Comput Graph 15(6):1121–1128
4. Bostock M, Ogievetsky V, Heer J (2011) D^3 data-driven documents. IEEE Trans Visual Comput Graph 17(12):2301–2309
5. Coene JP (2018) Sigmajs: an R htmlwidget interface to the sigma. js visualization library. J Open Sour Softw 3(28):814
6. Franz M, Lopes CT, Huck G, Dong Y, Sumer O, Bader GD (2016) Cytoscape. js: a graph theory library for visualisation and analysis. Bioinformatics 32(2):309–311
7. Guzzi PH, Roy S (2020) Biological network analysis: trends. Graph theory, and algorithms: Elsevier, Approaches
8. Jain A (2014) Data visualization with the d3. js javascript library. J Comput Sci Colleges 30(2):139–141
9. Nath K, Roy S (2019) Detecting intrinsic communities in evolving networks. Soc Netw Anal Mining 9(1):1–15
10. Nath K, Roy S, Nandi S (2020) Inovin: a fuzzy-rough approach for detecting overlapping communities with intrinsic structures in evolving networks. Appl Soft Comput 89:106096

A Novel Optimized Revenue Scheme in Finite Capacity SLA Aware Service Model in Fog Computing Environment

S. Panigrahi, Sudhanshu Shekhar Patra, Saeed Al-Amodi, and Rabindra K. Barik

Abstract Cloud computing is a pay-per-usage service provider offering shared resources through the Internet. Due to the latency issues, there is a fog layer in between the IoT devices and the cloud layer. For the fog provider, the major concern is to retain the impatient customers into the system and simultaneously maintain the SLA. This paper provides a finite buffer M/M/c/K queueing model through which the revenue losses being estimated for the fog system and the behavior of the impatient customers are studied. A profit function is suggested through which the loss incurred due to the system blocking is evaluated. Various performance measures of the system such as waiting time and loss probability are evaluated. The policy helps the fog service providers in taking the decision of how many servers should be active in the system, and what will be the capacity of the system to generate maximum profit.

Keywords Fog computing · Queueing model · Profit optimization · Blocking probability · Abandonment

1 Introduction

Cloud computing is now a widely used computer model because of its scalability, elasticity, and pay as per usage behavior [1]. Many industries are using the software, platform, and services from many cloud providers on pay-per-usage basis. But due to

S. Panigrahi · S. Al-Amodi
School of Computer Engineering, KIIT Deemed to be University, Bhubaneswar, India
e-mail: ctcsunil@gmail.com

S. Al-Amodi
e-mail: mrsaeedamodi@gmail.com

S. S. Patra (✉) · R. K. Barik
School of Computer Applications, KIIT Deemed to be University, Bhubaneswar, India
e-mail: sudhanshupatra@gmail.com

R. K. Barik
e-mail: rabindra.mnnit@gmail.com

the latency issue, many industrial applications can't be solved using cloud computing and so edge computing and fog computing are evolved and growing popularity in the industry [2]. Before taking the service from the service provider, there will be a service level agreement (SLA) between the provider and the consumer and signed by both parties. Quality of Service (QoS) is a SLA negotiation that specify the customer's expectation. Since no system is infinite in buffer capacity, a finite buffer capacity can be modeled to study the performance of such fog systems [3, 4]. In two situations, there will be a revenue loss in the system. First, due to customer abandonment where the customer leaves the system without waiting for service because of latency or long waiting queue length. Second, the queueing length can be minimized by reducing the waiting buffer, but in this case the client requests will be rejected most often result in revenue losses to the cloud provider.

In this paper, we optimized the profit in a finite capacity abandonment fog system. The blocking and abandonment loss are considered and estimated the revenue. A profit function is computed, and various optimized parameters are evaluated. The rest of the paper is organized as follows. Section 2 gives the previous researches done in this area, Sect. 3 shows the proposed multi-server fog model, Sect. 4 shows the cost analysis of our proposed model, in Sect. 5 the simulation results are shown, and finally in Sect. 6 conclusion with future endeavors.

2   Related Work

The general fog computing environment is illustrated in Fig. 1 [5]. In fog computing environment, there are many finite capacity queueing models have been studied by many researchers in the past. Goswami et al. [6, 7] have proposed the single server
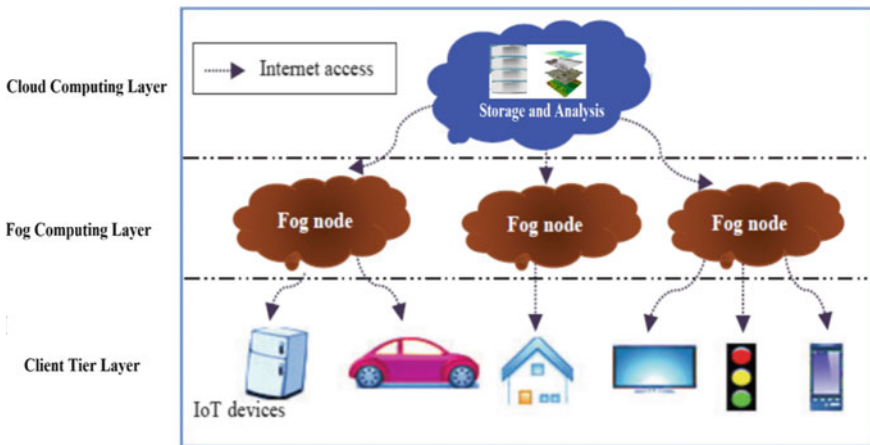
Fig. 1 General Fog computing environment with cloud, fog, and client tier layers

finite capacity rate control system to manage the service rate dynamically on the basis of the state of the current system. In [8], an analytic cost model was presented by considering the buffer size and the processor speed. The costs of customer loss and the client request delays have been studied. In [9–11], the authors have studied the performance of the system and the best resource utilization in a cloud setting with a finite population. Though there were many research works have been done in cloud computing, fog and edge computing, less works have been done in the buffer capacity restriction.

Barik et al. [12, 13] have proposed the model for maximum resource utilization and profit. In that authors have considered M/M/m queueing model and analytically solved for the best multi-server configuration to maximize the profit in the cloud system. In [13, 14], the system is modeled as M/M/c+ D in a homogeneous environment as a multi-server system. A double-quality-guaranteed renting system is developed to ensure profit maximization. The authors considered a multi-server queue to model the system and computed the various cloud system performance measures were used to scale up and down resources based on queue length and waiting time in the cloud system.

3   A Multi-server Model

In a multi-server model, multiple VMs are there which are deployed to give service in response to client requests. When the requests arrived to a fog system and all of the virtual machines are in use in solving the other requests, the requests by the clients waits in the waiting buffer.

3.1   M/M/c/K Queueing Model

We assume a fog layer have finite waiting buffer and can be modeled as M/M/c/K queueing model. There is c homogeneous VMs in the fog system, and maximum of K requests are allowed to the fog system including the requests which are currently in service. The client requests arrive from an infinite source by following a Poisson distribution with a mean arrival rate λ, and the service times are independently and identically distributed with exponential distribution μ. The system follows FCFS service discipline. The state transition for the arrival and processing of the client requests are shown in Fig. 2.

The arrival rate and the service rate can be written as:

$$\lambda_i = \begin{cases} \lambda, & 0 \leq i \leq K - 1 \\ 0, & i \geq K \end{cases} \tag{1}$$
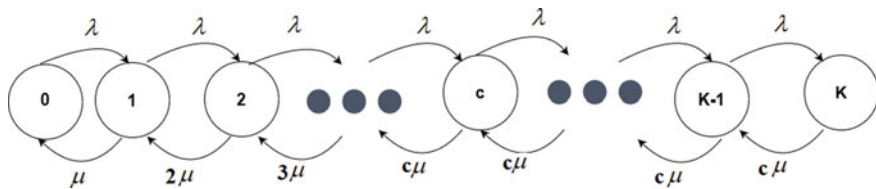
Fig. 2 State transition diagram for arrival and processing of the various request accepted by clients

$$\mu_i = \begin{cases} i\mu, & 1 \leq i \leq c - 1, \\ c\mu, & c \leq i \leq K \end{cases} \tag{2}$$

We define P_i as the probability that there are i client requests in the fog system at any given time. For a finite buffer multi-server queueing system with c homogeneous VMs, the steady-state distribution is given by

$$P_i = \frac{\lambda_0.\lambda_1.\lambda_2 \ldots \lambda_{i-1}}{\mu_1 \mu_2 \mu_3 \ldots \mu_i} \cdot P_0 = \frac{\lambda^i}{i!\mu^i} P_0, \; 1 \leq i \leq c - 1 \tag{3}$$

and

$$P_i = \frac{\lambda^i}{c!c^{i-c}\mu^i} P_0, \; c \leq i \leq K \tag{4}$$

Equations (3) and (4) are the closed form of P_i. To obtain P_0, we can apply the normalizing condition $\sum_{i=0}^{K} P_i = 1$.

$$\sum_{i=0}^{K} P_i = P_0 \cdot \left(1 + \sum_{i=1}^{c-1} \frac{\lambda^i}{i!\mu^i} + \sum_{i=c}^{K} \frac{\lambda^i}{c!c^{i-c}\mu^i} \right) \tag{5}$$

Then, the steady-state probability of zero client request in the fog system is

$$P_0 = \left(\sum_{i=1}^{c-1} \frac{\lambda^i}{i!\mu^i} + \sum_{i=c}^{K} \frac{\lambda^i}{c!c^{i-c}\mu^i} \right)^{-1} \tag{6}$$

3.2 Proposed Fog Control System

The proposed fog network is shown in Fig. 3 with finite capacity (represented by K). The system blocking rate and the percentage of users that abandon are depends on the capacity of the system along with VM provisioning control. User requests are
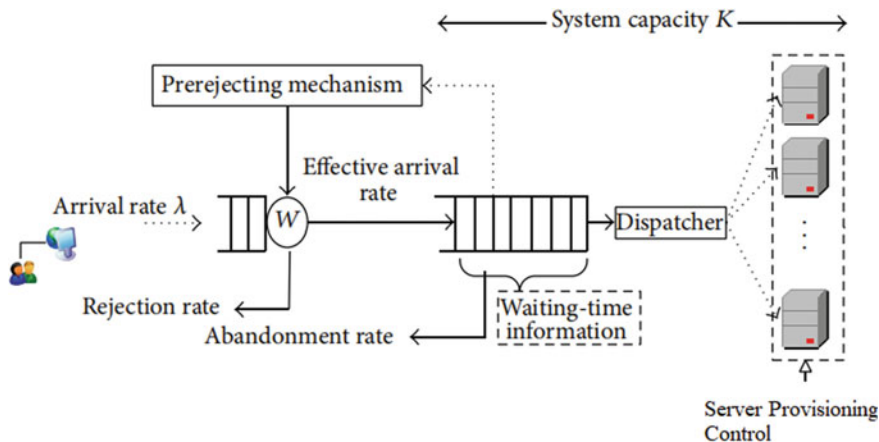
Fig. 3 A fog service model with server provisioning control

processed in the system if the VM is free or wait in the waiting buffer. If the waiting buffer is also full, then the request is lost by a defined pre-rejection mechanism (PRM). It's reasonable to believe that the customer requests are rejected before they came to the fog system the provider will not be penalized and the provider will not be responsible to give compensation to the customer. Therefore, rejection is avoided. The blocking probability is the fraction of rejecting arrival client requests and is denoted by P_K.

This can be derived by $i = K$ in Eq. (4) as follows:

$$P_K = \frac{\lambda^K}{c!c^{K-c}\mu^K} P_0 = \frac{\rho^K}{c!c^{K-c}} P_0 \tag{7}$$

where $\rho = \frac{\lambda}{\mu}$.

The effective arrival rate λ^e into the system is differs from the overall arrival rate and is determined by:

$$\lambda^e = \lambda(1 - P_K) \tag{8}$$

Client request after being rejected may join the system later as a new request after elapsing a random time.

3.3 Abandoned Rate

An accepted client request will be sent to the waiting buffer when all the VMs are busy and must wait one of the VM gets idle. The expected queueing length can be

computed as:

$$L_q = \sum_{i=c}^{K} (i - c) P_i \tag{9}$$

Using Little's formula, we can predict the waiting time W_q in the waiting buffer and is derived as:

$$W_q = \frac{L_q}{\lambda} \tag{10}$$

Balking and Reneging are the two queueing behavior which depend on waiting time and mainly affects the client's decision. Therefore, the system provider will note the rate of abandonment and update this to study the severity of the abandonment rate.

It is possible to calculate the Potential Abandonment Index as:

$$ai(s, t) = \frac{\overline{\lambda_d}}{W_q \lambda^e} \tag{11}$$

where $\overline{\lambda_d}$ is the mean rate of abandonment for a specific service pattern s ($s \in S$) at a period t ($t \in T$).

The abandonment probability

$$P_d = W_q . ai(s, t) = \frac{\lambda_d}{\lambda^e} \tag{12}$$

3.4 Probability of Loss

Client decides to accept a service with a probability of $1 - P_d$, or abandon the service with probability P_d. The expected abandonment rate is:

$$\lambda_d = \lambda^e . P_d = \lambda(1 - P_K).W_q, ai(s, t) \tag{13}$$

The clients those really want to take the service has the arrival rate

$$\begin{aligned} \lambda^* = \lambda^e - \lambda_d &= \lambda^e - \lambda^e . P_d \\ &= \lambda^e (1 - P_d) \\ &= \lambda(1 - P_K)(1 - P_d) \end{aligned} \tag{14}$$

Finally, the probability of loss can be computed as:

$$\text{Probability of Loss} = \frac{\lambda - \lambda^*}{\lambda}$$
$$= P_K + P_d - P_K.P_d \tag{15}$$

4 Cost Analysis

4.1 Revenue Cost

The total revenue Rev(c, K) per unit time for c VMs and K waiting buffer is given by:

$$\text{Rev}(c, K) = \lambda^e (1 - P_r) N(s, t)$$
$$= \lambda (1 - P_K)(1 - P_r) N(s, t) \tag{16}$$

$N(s, t)$ is the revenue/task; s is the service pattern ($S = 1, 2, 3 \dots s$); t is the period ($T = 1, 2, 3, \dots t$).

4.2 Power Usage Cost

$$\text{Pow}(c, \mu) = \frac{\varepsilon c (\varphi (\mu - \mu_b)^v + B)}{\rho^*} \tag{17}$$

Here μ_b is the base line service rate, B is the power usage by the baseline execution rate, ρ^* is the system utilization, ε is the cost per watt/unit time, and $\varphi = a.C.y^2/z^{2x+1}$.

4.3 Congestion Cost

The system congestion cost with c VMs and K waiting buffer is:

$$\text{Con}(c, K) = C_W W^* + C_H L^* + C_U (K - c) \tag{18}$$

where C_W = cost used by client request to wait in waiting buffer per unit time, C_H = cost used to hold client requests in waiting buffer per unit time, C_U = cost used by the client requests to wait in the waiting buffer per unit time, W^* = mean waiting time, L^* = Mean Queueing length.

4.4   VM Provisioning Cost

Let C_c be the VM provisioning cost/VM per unit time, then the VM provisioning

$$\text{cost} = \text{Prov(c)} = \frac{c.C_c}{\rho^*} \tag{19}$$

4.5   Profit Function

The objective is to find the optimum values for c^* along with K^* for a given service rate μ for maximizing the profit. Maximize Profit is a possible solution to the problem.

$$\text{Subject to } 0 \leq \mu_b < \mu$$
$$\text{Loss probability } < \text{ T} \tag{20}$$

where

$$\text{Profit} = f(c, K) = \text{Rev}(c, K) - \text{Prov}(c) - \text{Pow}(c, \mu) - \text{Con}(c, K) \tag{21}$$

5   Numerical Results

Experiments are conducted to study the validity of the fog system. The programs are written in MATLAB to validate the system. Simulations are done by taking the following system parameters:

$$\varepsilon = 0.1, \ B = 8, \ \lambda = 4200/\min, \ \mu = 50/\text{s}, \ \mu_b = 45/\text{s},$$
$$C_U = 32, \ C_R = 820, \ C_H = 65, \ C_W = 65, \ ai(s, t) = 0.01, \ T = 0.01$$

Profit distribution with various VMs and system capacities is shown in Table 1. From the table, it can be shown that in the beginning the profit increases when the system capacities and the number of VMs increases, but as it goes on increasing there will be no further increase in profit and it decreases gradually. The maximum profit of 1.094 E+06 is found at the optimal solution $(c^*, K^*) = (104, 130)$. Table 2 gives the loss probability for system capacities and number of VMs. From table, the loss probability decreases while the VMs increases. When the VMs increases the loss probability decreases and is more effective than system capacities. Figure 4

Table 1 Profit with system capacities and number of VMs

No. of VMs→ Sys. capacity↓	100	102	104	106	108	110
150	1,082,533	1,087,421	1,091,219	1,089,272	1,088,382	1,087,527
145	1,082,611	1,088,212	1,092,627	1,090,313	1,088,928	1,087,936
140	1,082,712	1,089,323	1,092,829	1,090,838	1,089,202	1,088,383
135	1,083,123	1,090,132	1,093,245	1,091,839	1,090,283	1,089,373
130	1,084,732	1,091,393	**1,094,322**	1,092,738	1,091,839	1,090,838
125	1,085,253	1,089,322	1,093,244	1,090,383	1,089,393	1,088,312
120	1,086,021	1,088,237	1,092,103	1,089,383	1,088,393	1,083,211

From the table, one can see that the maximum profit is earned with system capacity= 130 and number of VMs=104.

Table 2 Loss probability constraint with system capacities and number of VMs with a $T = 0.01$

No. of VMs→ Sys. capacity↓	100	101	102	103	104	105	106	107
140	0.0080	0.0052	0.0040	0.0032	0.0025	0.0018	0.0012	0.0008
139	0.0081	0.0053	0.0041	0.0033	0.0026	0.0019	0.0013	0.0009
138	0.0082	0.0054	0.0042	0.0034	0.0027	0.0020	0.0014	0.0010
137	0.0083	0.0055	0.0044	0.0035	0.0028	0.0021	0.0015	0.0011
136	0.0086	0.0056	0.0045	0.0036	0.0029	0.0020	0.0016	0.0012
135	0.0089	0.0065	0.0058	0.0047	0.0036	0.0031	0.0023	0.0021
134	0.0092	0.0082	0.0062	0.0056	0.0047	0.0038	0.0029	0.0028

displays the QoS improvement percentage for our proposed model including waiting time, reneging rate, blocking rate, and loss probability.

6 Conclusion

Any good system can be developed when the performance evaluations are done accurately and effectively the system controls can be done and so also in fog systems. Revenue and profit enhancement simultaneously following the SLA constraint is a major interest for the fog service providers. In this paper, we established the relationship between the client request abandonment on the loss probability with the system control. The system capacities, the utilization percentage and its effect on various performance parameters such as waiting time and loss probability are derived. The main goal of the paper is to maximize the profit of the system, and the revenue is calculated by considering the blocking loss, loss due to abandonment and the final arrival rate. Through numerical results the validity of the system is evaluated. In future, the work can be extended during the fault in the system and catastrophe situation having balking and reneging.
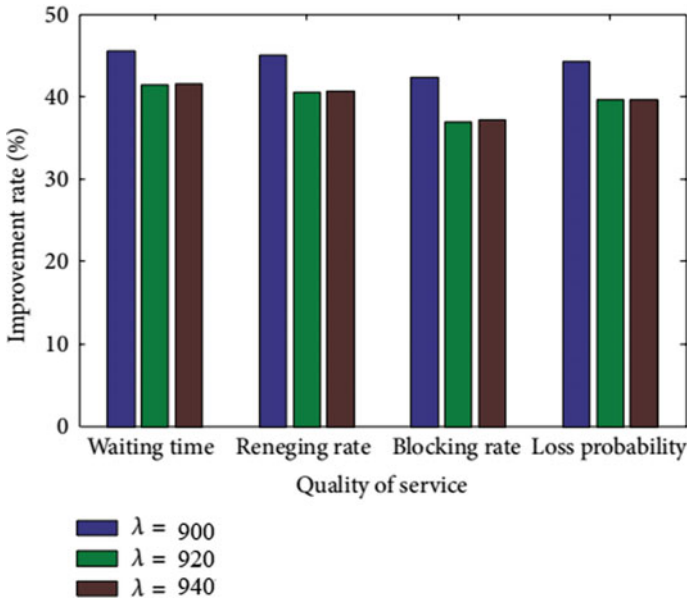
Fig. 4 QoS improvement values for various arrival rates

References

1. Buyya R, Yeo CS, Venugopal S, Broberg J, Brandic I (2009) Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility. Futur Gener Comput Syst 25(6):599–616
2. Singh J, Singh P, Gill SS (2021) Fog computing: a taxonomy, systematic review, current trends and research challenges. J Parallel Distrib Comput
3. Ghosh AP, Weerasinghe AP (2010) Optimal buffer size and dynamic rate control for a queueing system with impatient customers in heavy traffic. Stochast Proces Their Appl 120(11):2103–2141
4. Doran D, Lipsky L, Thompson S (2010) Cost-based optimization of buffer size in M/G/1/N systems under different service-time distributions. In: 2010 Ninth IEEE international symposium on network computing and applications, pp 28–35. IEEE
5. Goswami V, Patra SS, Mund GB (2012) Performance analysis and optimal resource usage in finite population cloud environment. In: 2012 2nd IEEE international conference on parallel, distributed and grid computing, pp 679–684. IEEE
6. Goswami V, Patra SS, Mund GB (2012) Performance analysis of cloud computing centers for bulk services. Int J Cloud Appl Comput (IJCAC) 2(4):53–65
7. Goswami V, Patra SS, Mund GB (2014) Quality of services and optimal management of cloud centers with different arrival modes. In: Multidisciplinary perspectives on telecommunications, wireless systems, and mobile computing, pp 185–198. IGI Global
8. Patra SS (2018) Energy-efficient task consolidation for cloud data center. Int J Cloud Appl Comput (IJCAC) 8(1):117–142
9. Khazaei H, Misic J, Misic VB (2011) Performance analysis of cloud computing centers using m/g/m/m+ r queuing systems. IEEE Trans Parallel Distrib Syst 23(5):936–943

10. Goswami V, Patra SS, Mund GB (2012) Performance analysis of cloud with queue dependent virtual machines. In: 2012 1st international conference on recent advances in information technology (RAIT), pp 357–362. IEEE
11. Gourisaria MK, Khilar PM, Patra SS (2021) ESPS: energy saving power spectrum-aware scheduling to leverage differences in power ratings of physical hosts in datacenters. Informatica 45(6)
12. Barik RK, Patra SS, Kumari P, Mohanty SN, Hamad AA (2021) A new energy aware task consolidation scheme for geospatial big data application in mist computing environment. In: 2021 8th international conference on computing for sustainable global development (INDIACom), pp 48–52. IEEE
13. Panigrahi SK, Barik RK, Behera S, Barik L, Patra SS (2021) Performability analysis of FogGIS model for geospatial web services. In: 2021 11th international conference on cloud computing, data science & engineering (confluence), pp 239–243. IEEE
14. Rout S, Patra SS, Mohanty JR, Barik RK, Lenka RK (2021) Energy aware task consolidation in fog computing environment. In: Intelligent data engineering and analytics, pp 195–205

Analysis of COVID-19 Data Through Machine Learning Techniques

Nayak Padmalaya, Trivedi Veena, and Jugge Praveen

Abstract This pandemic environment of COVID-19 is a global problem that requires deeper research to analyze and predict the impact on humans soon. It is an infectious disease activated by SARS-CoV-2 that can affect the human upper respiratory tract like sinus, nose, throat, and lower respiratory tracks like windpipes, lungs, etc. At present, the non-availability of proper medication insufficient vaccination creates a panic mode across the world, and the disease is spreading exponentially day by day in all countries as well as India. This lay emphasis on humans staying at home as a preventative measure to protect against COVID-19. However, people sealed at home cannot be treated as safe if one person goes out for emergency work. This research paper aims to study the symptoms of a COVID-19 patient and to predict the health condition of a patient using the Fuzzy Logic model. Furthermore, this study aims to analyze the current COVID-19 cases in India and to forecast the number of positive, mortality, and recovered cases for the next few months through various machine learning techniques such as AutoRegression, MLP Regression, Linear Regression, and SVM Regression based on the Kaggle dataset. Our experimental results show that Autoregression produces better accuracy than other regression models.

Keywords COVID-19 · Autoregression · MLP regression · Linear regression · SVM regression

N. Padmalaya
Department of CSE, Gokaraju Lailavathi Womens Engineering College, Hyderabad, Telangana, India
e-mail: padmalaya@griet.ac.in

T. Veena (✉)
Computer Engineering Department, VESIT, Mumbai, India
e-mail: veena.trivedi@ves.ac.in

J. Praveen
Department of EEE, GRIET, Hyderabad, Telangana, India
e-mail: drpraveen@griet.ac.in

1   Introduction

In the present day's scenario, COVID-19 or the novel coronavirus is the most focused topic being discussed among human beings, among the news channels, across social media, among the researchers, and so on. According to a different coronavirus epidemic caused by the severe acute respiratory syndrome coronavirus (SARS-CoV-1) virus had appeared in Guangdong, China, and around 26 countries were affected [1–3]. In December 2019, a pneumonia outbreak coincided with the spring festival of China and rapidly spread across the country. Based on recent literature studies, government documents, up-to-date Internet base studies, and WHO's information, it is observed that the epidemiology, clinical manifestations, virology and origin, pathology, and treatment of COVID-19 infection, partial resembled with (SARS-CoV-1) and Middle East respiratory syndrome coronavirus (MERS-CoV) infection indicating a bat origin [4]. The Chinese Centre for Disease Control and Prevention (CCDC) recognized a novel beta-coronavirus called 2019-nCoV, now officially familiar as SARS-CoV-2, that was the origin for the pandemic [5]. Although early studies bring a link between single local fish and wild animal market, specifying the possible infection between animal-to-human transmission, studies have increasing proofs of human-to-human transmission of SARS-CoV-2 via droplets or direct contacts. This was the most infectious disease in the first two decades of the twenty-first century took place for the third time in China, letting human-to-human transmission and alarming global health concerns [6]. Awareness of the biological features of 2019-nCoV needs to be systematically summarized and updated from time to time to help the optimized control measures. As of August 15, 2020, the pandemic caused an accumulation of 21,357,890 confirmed cases including 763,387 death cases and 14,151,766 recovered cases all over countries worldwide. This pandemic environment of COVID-19 is a global concern that requires deeper research to analyze and predict the impact on humans soon. India is the second populated country in the world. The administrative system itself is not strong enough to handle this pandemic situation. Various options like immunization, quarantine, and lockdown are being opted, yet a lot more is required [7].

The structure of the paper is given as follows. Section 2 discusses the current literature, COVID-19 symptoms, and uses a Fuzzy Logic model to predict the depth of infection of a patient. Section 3 discusses the current COVID-19 cases in India and presents comparative predictions for the next month through various machine learning models. Section 4 concludes the paper.

2   Background Discussion

Data mining and big data techniques are playing a crucial role in medical research for a long. AI has been applied to different types of data available at the National and International levels to detect, prevent, and predict to fight against the COVID-19

pandemic [8–11]. This paper has referenced a few machine learning algorithms that have been applied to the COVID-19 dataset for the eradication of this pandemic situation across the globe. The effectiveness of the prediction depends upon the quality of the data source and the predictions may vary w.r.t. the impurities of data sources. In [12], the authors have predicted the spread rate of novel coronavirus considering environmental factors like wind speed, humidity, and temperature accessing the local weather database and WHO database. The research work in [4] discusses a decision-making scheme by analyzing the coronavirus data fetching from Johns Hopkins University. In [13], the authors of the paper have analyzed the dataset from the Centres for Disease Control (CDC), US and demonstrated that the spread rate can be reduced by disease control interventions and movement restrictions. The authors of [14–17] have explained the impact of quarantine to reduce the spread rate of COVID-19. The research result predicts the mortality rate by fetching the COVID-19 databases from Italy national data, WHO database, and Johns Hopkins University, respectively [1, 3, 18–21]. ML has long been acknowledged by many researchers as a standard software tool for modeling natural catastrophes and climate forecasting [22, 23]. Even if many ML methods such as random forest, Bayesian networks, Naïve Bayes, neural networks, genetic programming, classification, decision tree, etc., are applied for various applications, the application of ML in the COVID-19 outbreak is still in the initial stage [24–27]. More sophisticated hybrid ML methods must be assembled and need to be explored. In this context, we aim to investigate the simplified ability of the proposed ML models and to predict the accuracy of the pandemic environment. Our contributions are twofold as given below.

- Understanding the symptoms of COVID-19 patients and predicting the health condition using the FL model through the MATLAB simulation tool.
- Processing the COVID-19 data (India) related to the number of confirmed cases, the number of death cases, recovery cases, and predicting the number of related cases with an accurate model. The dataset is fetched from the Kaggle database. The results confirm that the Autoregression model can be treated as an approximation forecast model.

2.1 COVID-19 Symptoms

The primary symptoms of COVID-19 have been referred to from the Letco database [28]. These symptoms are listed as; (i) Sneezing, (ii) runny nose, (iii) cough, (iv) diarrhea, (v) fever and pain, (vi) sore throat, and (vii) exacerbated asthma. Usually, flu or cold-like symptoms can be cured within 3–5 days, but COVID-19 takes up to 14 days or more for the cure. However, these symptoms are not constant for all and vary from person to person. Researchers and doctors across the country are still working to discover the antidrug of COVID-19. World Health Organization (WHO) has listed a "do and don't do list" for the public guidelines [1]. These are (i) to consult a doctor at an early stage, (ii) to avoid the public place, (ii) to drink sufficient warm water, (iii) not to join in any gathering, (iv) to take proper medicine for fever and

pain, (v) to use a clean steam vaporizer, (vii) to take precautions while sneezing, (viii) avoid smoking and, (ix) washing hands very often with sanitizer, (x) use a face mask, and (xi) maintain social distancing. So, it is very essential to study and analyze the spreading rate, structure, and determination of health and disease conditions for a huge, populated country like India. Isolation, quarantine, travel restrictions, and social distancing are the primary steps to fight against coronavirus.

2.2 Prediction of Patient's Health Condition Using Fuzzy Logic (FL) Model

The term fuzzy tends to vague things that make things unclear. In the real-world scenario, many times we face some situations where the state cannot be determined whether it is true or false. In the Boolean system, 1 and 0 can be represented as absolute truth and false, respectively. But in the fuzzy system, no state can be defined by 0 or 1, rather many intermediate states represent partially true and partially false. Fuzzy Logic can be applied where the information is beyond control, imprecise, or without proper boundaries. In this way, the inaccuracies and uncertainties of any situation can be considered. Fuzzy Logic-based computing methods are used in the development of intelligent systems for pattern recognition, decision-making, optimization, and control [29]. Almost all real-life applications based on Fuzzy Logic use linguistic variables whose values are represented by words rather than numbers and use a fuzzy if–then rule or, simply, called a fuzzy rule. In our prediction, we have considered 4 linguistic variables such as sneezing, dry cough, diarrhea, high fever, and sore throat, and exacerbated asthma to predict the probable condition of a patient. The proposed model is simulated through MATLAB Fuzzy Toolbox as shown in Fig. 1. The fuzzy model toolbox is depicted in Fig. 2. Each variable is
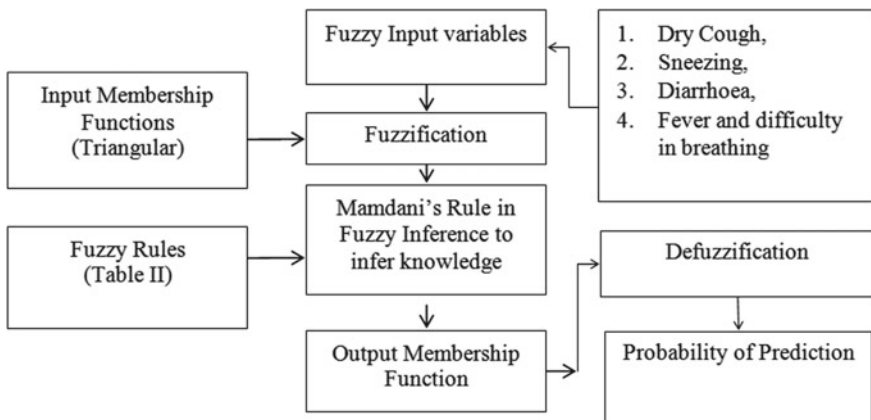
Fig. 1 Proposed Fuzzy Logic model to predict a patient's health condition
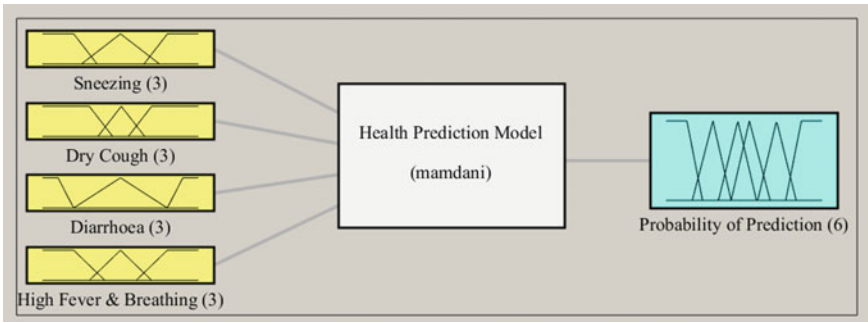
Fig. 2 Proposed health prediction model using fuzzy toolbox

represented by a triangular membership function as depicted in Fig. 3. The output membership function is shown in Fig. 4. We have represented r as linguistic variables such as $r = 4$. Each variable considers 3 conditions such as $n = 3$. The total number of rules is derived based on a mathematical formula given in Eq. 1. The linguistic values for each variable are defined as High, Medium, and Low. We have represented the levels of each variable in a measuring scale as High $= 5$, Medium $= 3$, and Low
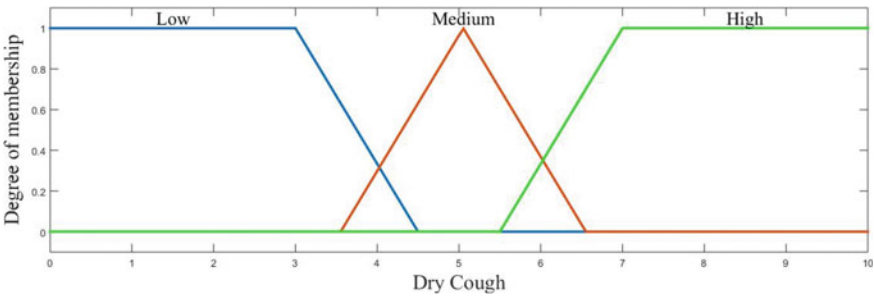
Fig. 3 Input membership function for dry cough
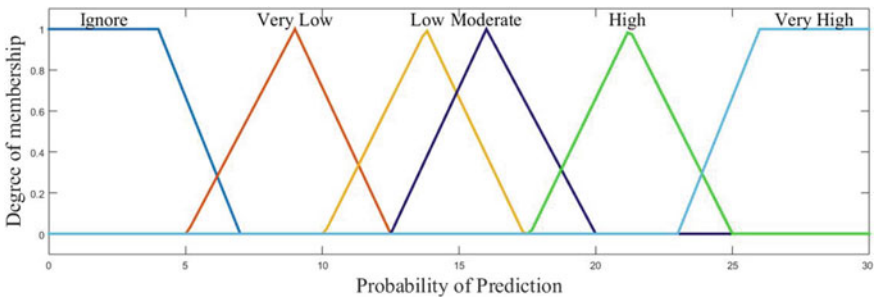
Fig. 4 Output membership function

= 1 for input variables as shown in Table 1. Similarly, we have represented the output variables in a range Very High = 11–20, High = 9–10, Moderate = 7–8, Very Low = 5–6, and Neglect = 4. Mamdani's rule is applied for fuzzification as given in Table 1. The input variables are shown in Table 2. After fuzzification, the output is fed for defuzzification using a linear interpolation technique. The probability of the health condition of a patient is shown in Fig. 8 obtained from the fuzzy toolbox. Table 3 describes the health condition of a patient in a tabular form referring to Fig. 5.

$$T_{\text{rules}} = n^n \times (r - 1) = 3^3 \times 3 = 81 \tag{1}$$

2.3 Forecasting COVID-19 Cases in India Using Machine Learning Models

The first COVID-19 case in India was confirmed by the Government of India on January 30, 2020, when a student traveled back from Wuhan to the Kerala state. As the infection cases started increasing slowly, the Government of India had taken a few precautionary measures to defend against COVID-19. On 22 March "Janata Curfew" was followed by all citizens of India as per the instructions given by Govt. of India on March 19, 2020. The Government of India again announced a nationwide lockdown for 21 days on March 24, 2020. The entire population of India (approximately 1.3 billion) observed nationwide lockdown as a precautionary measure against the COVID-19 pandemic and could limit the number of positive cases to approximately 500 by March 2020. The forced lockdown had resulted in a slower growth rate of the pandemic to a rate of doubling every six days by 6 April. Even, it could reduce the rate of doubling every eight days by 18 April. On April 14, 2020, the nationwide lockdown was extended until May 3, 2020 followed by some relaxation service after April 20, 2020 (the minimal spread region). On May 1, 2020, the nationwide lockdown was further extended until May 17, 2020. All the areas are marked by the Government of India into three zones such as green, red, and orange with relaxations based on the number of infections. After May 17, 2020, the lockdown was extended till May 31, 2020. To help daily laborers, government passed the order for lockdown till June 30, 2020 in containment zones, with relaxed services starting from June 8, 2020, which is termed as Unlock 1. As of June 25, 2020 total, COVID-19 cases reached 4, 56,183 confirmed cases, reached 14, 476 2, mortality cases, and recovered 2, 58, 685. Due to the drastic economical dropout and to improve the economic status and to help different categories of people, the government opted for more relaxation and "Unlock 2" started from 1st July to 31st July. Even "Unlock 3 started from 1st August which provides more relaxation service. Figures 6 and 7 depict the COVID-19 cases in India till August 2020.

Table 1 Mamdani's if-else rules

Sl No.	Sneezing	Dry cough	Diarrhea	High fever and breathing	Probability of Prediction
1.	Low	Low	Less	Low	Ignore
2.	Medium	Low	Less	Low	Very Low
3.	High	Low	Less	Low	Very Low
4.	Low	Medium	Moderate	Medium	Moderate
5.	Medium	Medium	Moderate	Medium	Moderate
6.	High	Medium	Moderate	Medium	**High**
7.	Low	High	More	High	**High**
8.	Medium	High	More	High	**Very High**
9.	High	High	More	High	**Very High**
10.	Low	Low	Less	Low	Ignore
11.	Low	Medium	Less	Low	Very Low
12.	Low	High	Less	Low	Very Low
13.	Medium	Low	Moderate	Medium	Low
14.	Medium	Medium	Moderate	Medium	Moderate
15.	Medium	High	Moderate	Medium	**High**
16.	High	Low	More	High	**High**
17.	High	Medium	More	High	**Very High**
18.	High	High	More	High	**Very High**
19.	Low	Low	Less	Low	Ignore
20.	Low	Low	Moderate	Low	Very Low
21.	Low	Low	More	Low	Very Low
22.	Medium	Medium	Less	Medium	Moderate
23.	Medium	Medium	Moderate	Medium	Moderate
24.	Medium	Medium	More	Medium	**High**
25.	High	High	Less	High	**High**
26.	High	High	Moderate	High	**Very High**
27.	High	High	More	High	**Very high**
28.	Low	Low	Less	Low	ignore
29.	Low	Low	Less	Medium	Very low
30.	Low	Low	Less	High	Very low
31.	Medium	Medium	Moderate	Low	Low
32.	Medium	Medium	Moderate	Medium	Moderate
33.	Medium	Medium	Moderate	High	**High**
34.	High	High	More	Low	**High**
35.	High	High	More	Medium	**Very High**
36.	High	High	More	High	**Very High**
37. . 81					

Table 2 Different levels of I/P variables

Symptoms	Lower	Medium	High
Sneezing	1–2	3–4	5–10
Dry cough	1–2	3–4	5–10
Diarrhea	1–2	3–4	5–10
High fever and difficult in breath	97°–99°	100°–102°	103°–106°
Sore throat	0 (no)	1 (yes)	> 1 (severe
Exacerbated asthma	0	1 (yes)	> 1 (severe)

Table 3 Predicted output after defuzzification

I/P symptoms	Levels	Health prediction after defuzzification
Sneezing	5	16.2 (very high)
Dry cough	5	
Diarrhea	5	
High fever and difficult in breath	1 (yes)	
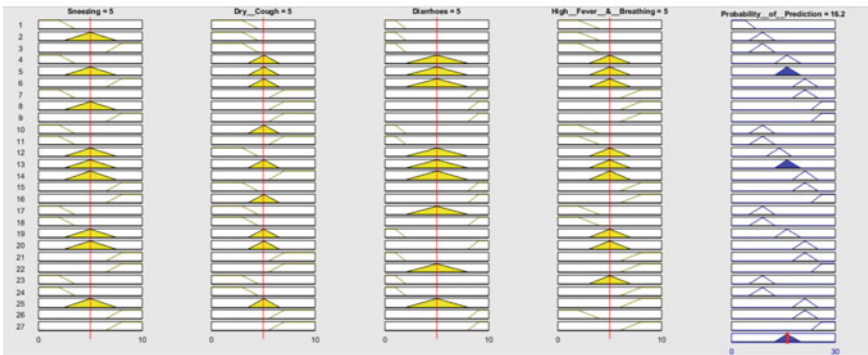
Fig. 5 Predictions (defuzzified output)

2.3.1 Proposed Data-Driven Model

In this section, we propose various machine learning models and explore an accurate prediction model that would forecast the growth of the infection for the upcoming month keeping the current dynamics of COVID-19 in mind. In our analysis, we have used various Machine Learning models such as the Autoregression model, MLP model, Linear regression model, and Support Vector Machine (SVM) model as depicted in Fig. 8. We have considered the number of cases in India for our analysis and three different scenarios such as the number of confirmed cases, the number of death cases, and the number of recovered cases because of a local outbreak. As discussed above, as of now, the total number of confirmed cases in India is
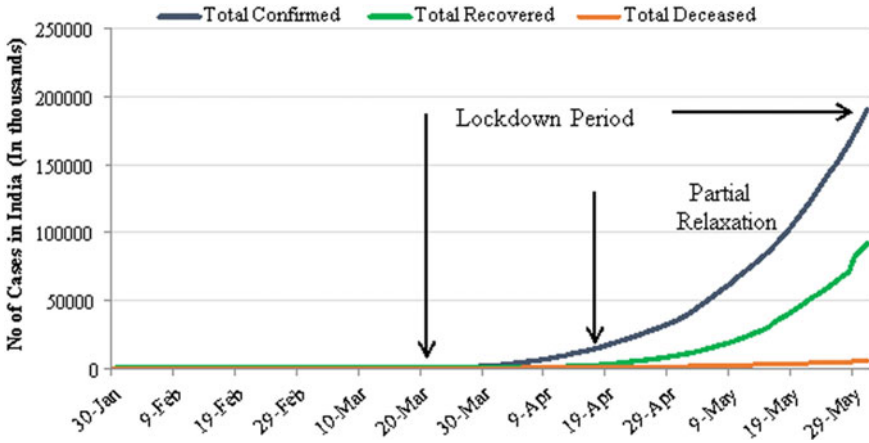
Fig. 6 COVID-19 cases starting from 1st day and during lockdown in India
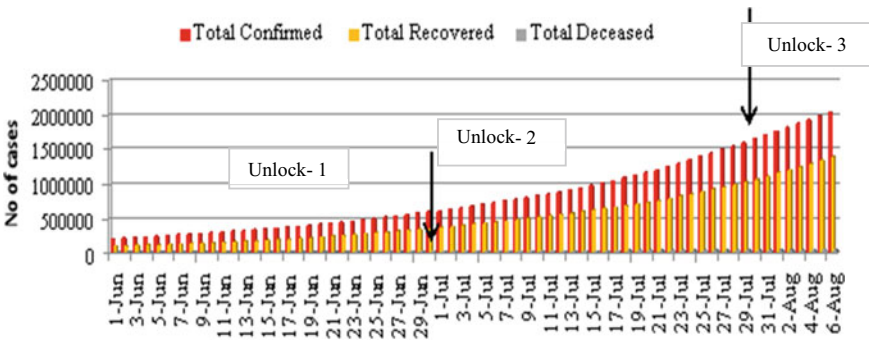
Fig. 7 COVID-19 in India after lockdown period
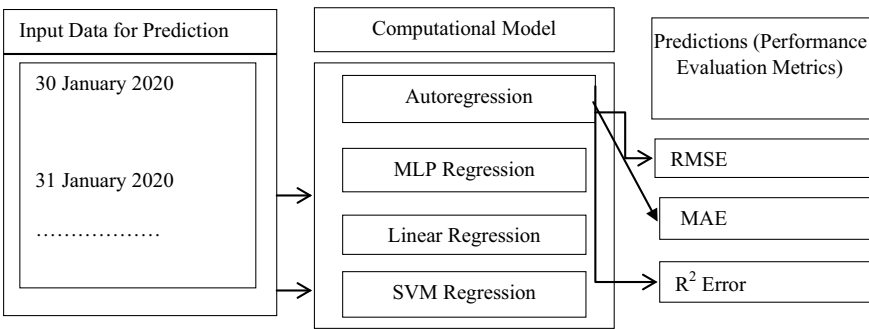
Fig. 8 Proposed computational model to forecast COVID-data

around 2,527,308, recovered cases 1,809,542, and death cases 49,148. The situation of COVID-19 is changing severely with each passing day and the data being gathered is huge in quantity with different formats. From the outcome of the study, it is expected that our predictions will surely help in boosting the resource allocation in each city during the upcoming days when the epidemic is crossing through serious public concern.

3   Experimental Results and Discussions

COVID-9 dataset for India is collected from the Kaggle database based on the confirmed cases, recovered cases, and deceased cases up to July 14, 2020, but trained the above-said models by considering only the data set up to July 20, 2020 to predict the status for the next one month. The reason is that we can measure the accuracy of the predicted model by taking the actual cases occurring daily. The following metrics are considered as the performance metrics to predict the number of cases for the upcoming one month to create awareness among the humans as well as to help the administrative control measure at the State as well as the National level. The same prediction can be applied to any country across the globe. We have considered Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-Square as the performance metrics to choose the best-predicted model. MAE is one of the significant performance metrics that represent the difference between the actual and predicted values attained by averaging the absolute difference in the given dataset. Statistically, it can be represented as shown in Eq. 7.

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^{N} |y_i - \widehat{y}| \tag{7}$$

where y_i represents actual values and $\widehat{y}$ represents the predicted values over a dataset containing N samples of data. The Root Mean Squared Error (RMSE) error rate is another performance metric that can be obtained by the square root of MSE used for numerical predictions. Mathematically, it can be demonstrated as shown in Eq. 8.

$$\sqrt{\frac{1}{N} \sum_{i=1}^{N} \left(y_i - \widehat{y}\right)^2} \tag{8}$$

where y_i represents actual values and $\widehat{y}$ represents the predicted values over a dataset containing N samples of data. R-squared (R^2) is a statistical measure that signifies the goodness of fit of a regression model. It is assumed that the best value for R-square is always 1. The more R-square value is close to 1, the best fit would be the model. Mathematically, it can be expressed as follows in Eq. 9.

$$R^2 = 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}} \tag{9}$$

where SS_{res} represents the residual sum of squares which is calculated by the summation of squares of perpendicular distance between data points and the best-fitted line. SS_{tot} represents the total sum of squares which is calculated by summation of squares of perpendicular distance between data points and the average line. From the actual confirmed cases, recovery cases, and the number of deceased cases, we have predicted the number of cases through various machine learning models. From these actual and predicted cases, we have calculated Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and R^2 score which is given in Table 4. We have used various machine learning models to forecast for the next 30 days for the Confirmed Cases, Recovered Cases, and Deceased Cases as fetched from the Kaggle database. Our findings conclude that the Autoregression model provides a near accurate model and less error rate w.r.t RMSE, MAE and provides better R^2 Score compared to MLP Regression, Linear Regression, and SVM Regression Model as plotted in Fig. 9.

Table 4 Different error rates for confirmed cases, recovery cases, and deceased cases

Different error rates for COVID-19 confirmed cases			
Model/metrics	Root mean square error	Mean absolute error	R^2 score
Autoregression	1411.475	1138.950	0.976
MLP regression	191,359.20	144,445.43	0.334258
Linear regression	524,244.62	477,296.84	−3.996604
SVM regression	718,990.79	679,661.89	−8.398393
Different error rates for COVID-19 recovered cases			
Model/metrics	Root mean square error	Mean absolute error	R^2 score
Autoregression	2607.763	1875.143	0.832648
MLP regression	144,854.072	113,327.81	0.180325
Linear regression	355,801.46	321,773.55	−3.945336
SVM regression	445,183.16	415,435.62	−6.742082
Different error rates for COVID-19 deceased cases			
Model/metrics	Root mean square error	Mean absolute error	R^2 score
Autoregression	251.587	97.563	0.000372
MLP regression	3920.65	3589.82	0.326767
Linear regression	13,778.42	13,102.83	−7.314724
SVM regression	19,567.58	18,973.22	−15.769625
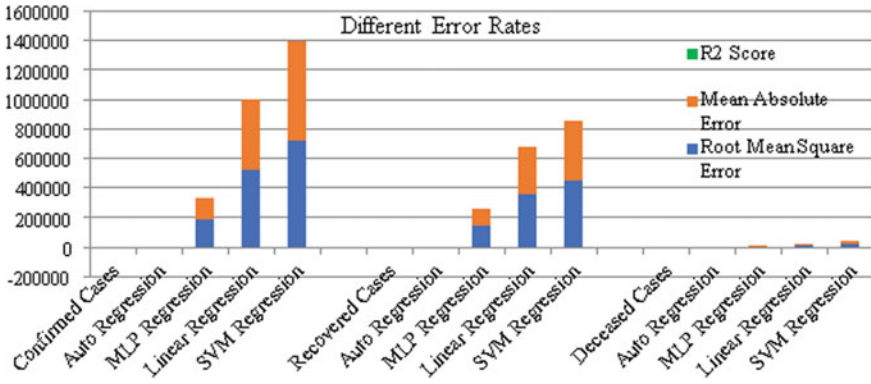
Fig. 9 Error rates for different regression techniques

4 Conclusion

Despite the whole world's efforts to understand COVID-19, many issues remain unclear. COVID-19 is a global problem and has affected almost all the whole world, irrespective of the country, state, gender, etc. The role of Artificial Intelligence is quite crucial to predict the patient's health status as well as predict the accurate model to take precautionary measures at the individual level as well as the administrative level. In this paper, we have used the Fuzzy Logic model to predict the health condition of a patient, and a few regression models to predict the status of COVID-19 cases that will be occurring near future. After a regressive trial, we found that the Autoregression model gives an accurate model with minimal error rate for COVID-19 data (India) and that can be used as a reference model to predict the advanced cases and precautionary measures can be employed at the National and International level.

References

1. Caccavo D (2003) Chinese and Italian COVID-19 outbreaks can be correctly described by a modified SIRD model. medRxiv. https://doi.org/10.1101/2020.03.19.20039388. Centers for disease control and prevention (CDC) 2020, Update: Outbreak of severe acute respiratory syndrome–worldwide, 2003, MMWR Morb Mortal Wkly Rep 52(17):388–391
2. Huang C, Wang Y, Li X, Ren L, Zhao J et al (2020) Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. The Lancet 395(10223):497–506
3. Zumla A, Hui DS, Perlman S (2015) Middle east respiratory syndrome. The Lancet 386(9997):995–1007. https://doi.org/10.1016/S0140-6736(15)60454-8
4. Toda A (2020) Susceptible-infected-recovered (SIR) dynamics of COVID-19 and economic impact. arXiv preprint arXiv:2003.11221
5. Wang L, Wang Y, Ye D, Liu Q (2019) Review of the 2019 novel coronavirus (SARS-CoV-2) based on current evidence. Int J of Antimicrob Agents 55(6):105948. https://doi.org/10.1016/j.ijantimicag.2020.105948

6. WHO Situation Report-94 Coronavirus Disease 2019 (COVID-19) (2020) Available online at: https://www.who.int/docs/default-source/coronaviruse/situation-reports/20200423-sitrep-94-covid-19.pdf?sfvrsn=b8304bf0_4. Accessed 10 Mar 2020

7. Irani T, Thi Nguyen HO, Marios K, Julian D et al (2020) Breadth of concomitant immune responses prior to patient recovery: a case report of non-severe COVID-19. Nat Med 26(4):453–455. https://doi.org/10.1038/s41591-020-0819-2

8. Bhatt C, Dey N, Ashour AS (eds) (2017) Internet of things and big data technologies for next-generation healthcare. Studies in Big Data 23. Springer International Publishing AG

9. Hassanien AE, Nilanjan D, Surekha B (eds) (2018) Medical big data and the internet of medical things: advances, challenges, and applications. CRC Press, Boca Raton

10. Arushi J, Vishal B (2017) Concoction of ambient intelligence and big data for better patient ministration services. Int J Ambient Comput Intell (IJACI) 8(4):19–30

11. Lan K, Wang DT, Fong S, Liu LS, Wong KKL et al (2018) A survey of data mining and deep learning in bioinformatics. J Med Syst 42(8):139

12. Bhattacharjee S (2003) Statistical investigation of relationship between spread of coronavirus disease (COVID-19) and environmental factors based on study of four mostly affected places of China and five mostly affected places of Italy. arXiv preprint arXiv: 2003.11277

13. Liu P, Beeler P, Rajan CK (2020) COVID-19 progression timeline and effectiveness of response-to-spread interventions across the United States. https://doi.org/10.1101/2020.03.17.20037770

14. Hossain M, Junus A, Zhu X, Jia P, Wen TH et al (2020) The effects of border control and quarantine measures on the global spread of COVID-19. medRxiv preprint. https://doi.org/10.1101/2020.03.13.20035261

15. Shahid Nadim SK, Indrajit G, Joydev C (2020) Short-term predictions and prevention strategies for COVID-2019: a model based study. arXiv preprint arXiv:2003.08150

16. Rocha F, dos Santos FSG, Gomes VB, Rocha TA, Croda JH et al (2020) Expected impact of COVID-19 outbreak in a major metropolitan area in Brazil. https://doi.org/10.1101/2020.03.14.20035873

17. Traini MC, Caponi C, De Socio GV (2020) Modelling the epidemic 2019-nCoV event in Italy: a preliminary note. medRxiv. https://doi.org/10.1101/2020.03.14.20034884

18. Siwiak MM, Szczesny P, Siwiak MP (2020) From a single host to global spread: the global mobility based modelling of the COVID-19 pandemic implies higher infection and lower detection rates than current estimates. https://doi.org/10.1101/2020.03.21.20040444

19. Teles P (2020) Predicting the evolution of SARS-Covid-2 in Portugal using an adapted SIR Model previously used in South Korea for the MERS outbreak. arXiv preprint arXiv:2003.10047.2020

20. Zareie B, Roshani A, Mansournia MA, Rasouli MA, Moradi G (2020) A model for COVID-19 prediction in Iran based on China parameters. medRxiv. https://doi.org/10.1101/2020.03.19.20038950

21. Bondugula RK, Udgata SK, Bommi N (2021) Machine learning model for COVID-19 infection classification using CT scan images. Arab J Sci Eng. https://doi.org/10.1007/s13369-021-05879-y

22. Mosavi A, Ghamisi P, Faghan Y, Duan P (2020) Comprehensive review of deep reinforcement learning methods and applications in economics. arXiv preprint http://arXiv:2004.01509

23. Nádai L, Imre F, Ardabili S, Gundoshmian TM, Gergo P et al (2020) Performance analysis of combine harvester using hybrid model of artificial neural networks particle swarm optimization. arXiv preprint arXiv:2002.11041

24. Liang R, Lu Y, Qu X, Li C, Xia S et al (2020) Prediction for global African swine fever outbreaks based on a combination of random forest algorithms and meteorological data. Transbound Emer Dis 2020 67(2):935–946. https://doi.org/10.1111/tbed.13424

25. Mezzatesta S, Torino C, De Meo P, Fiumara G, Vilasi A (2019) A machine learning-based approach for predicting the outbreak of cardiovascular diseases in patients on dialysis. Comput Methods Programs Biomed 177:9–15. https://doi.org/10.1016/j.cmpb.2019.05.005

26. Raja DB, Mallol R, Ting CY, Kamaludin F, Ahmad R et al (2019) Artificial intelligence model as predictor for dengue outbreaks. Malays J Public Health Med 19(2):103–108

27. Tapak L, Hamidi O, Fathian M, Karami M (2019) Comparative evaluation of time series models for predicting influenza outbreaks: application of influenza-like illness data from sentinel sites of healthcare centers in Iran. BMC Res. 12. https://doi.org/10.1186/s13104-019-4393-y
28. Letco Medical (2019) Master formula database. December 20, 2019. https://www.letcomedical.com/formulas
29. Padmalaya N, Anurag D (2016) Fuzzy logic-based clustering in wireless sensor networks to extend the network lifetime. IEEE Sens J 16(1)

Ubiquitous Healthcare System Using Recent ICT

Subasish Mohapatra , **Amlan Sahoo** , **Subhadarshini Mohanty** ,
and Prashanta Kumar Patra

Abstract Nowadays, healthcare is the most important domain to be analyzed by the technology experts for replacing the traditional techniques. Unexpected death is happening due to lack of medical treatment and diagnosis at right time. As healthcare applications have become technology-oriented, therefore, the upcoming technologies can be integrated for providing better, cheaper, faster solution. Internet of Things (IoT) has come up with the solution mechanism by real-time monitoring of the body parameters of the patients. It comprises different type of sensors to collect and transmit acquired data via Internet. This technology allows to integrate multiple sensors for providing real-time data. The proposed model illustrates a wearable sensor node which will be carried by the patients. The sensor nodes are responsible for monitoring the body parameters like temperature, pulse rate, etc., and transfer these data to the cloud via sink node. The sink node is connected to the cloud via an inbuilt Wi-Fi module. The sensor depicted data are stored in the cloud. An alert message will be sent to the concerned medical actors as well as to the patient for necessary aid when any abnormal signs are accessioned by sensor node.

Keywords IoT · Cloud server · Healthcare · Sensor · Sink node ·
Microcontroller · Wireless sensor network · Gateway

S. Mohapatra (✉) · A. Sahoo · S. Mohanty · P. K. Patra
Odisha University of Technology and Research, Bhubaneswar, Odisha, India
e-mail: smohapatra@cet.edu.in

S. Mohanty
e-mail: sdmohantycse@cet.edu.in

P. K. Patra
e-mail: pkpatra@cet.edu.in

1   Introduction

Healthcare is a most compelling area which requires continuous monitoring of body parameters such as temperature, respiratory rate, blood circulatory strain, heart rate, and blood-oxygen satiety. In the traditional healthcare architecture, there is no such provision available for real-time monitoring of the body parameters [1]. In this pandemic situation, it is not apposite to stand by in the queue and make arrangements for the pathological examination like traditional medical care framework along with steady checking of the well-being parameters. Therefore, remote health monitoring has been considered as a perfect solution-centric approach for the healthcare domain. It can be more practically empowered via the use of a promising technological paradigm known as Internet of Things (IoT). The IoT utilizes sensor and communication technologies along with pervasive and ubiquitous computing for upgrading corporeal objects into smart objects. Patients along with the sensors will act as the smart things in this modern technology [2]. This enables the delivery of ubiquitous and real-time healthcare services to the medical actors as well as the patients. IoT architecture involves three major layers: physical, network, and application. Smart objects are equipped with sensors to collect heterogeneous data. These sensors are limited by computational dimensions and life expectancy. Therefore, except data acquisition process, data processing and dissemination need to be done at the cloud end. The network layer is accountable for collecting and processing the acquired data. For better accuracy and precision, there is a need for collecting a large number of sensor data and a huge storage space. With the help of a large volume of sensor-acquired data, better decisions can be made. The network layer also facilitates the transparency between heterogeneous IoT objects. Further, the application layer helps to disseminate the result to a graphical user interface for easing real-time monitoring. Thus, the IoT innovation immensely expands the information system, execution, productivity, exactness, and speed of any real-time framework activity [3]. The utilization of sensor nodes has made information transaction, experimentation, and visualization productive, just as processing it in real-time. Also, a real-time health monitoring system can be thoroughly characterized as recurring or continuous observations of the patient's physiological state, and the capacity of life support apparatus, deliberately involving health administrators' decisions.

So remote patient monitoring and controlling is the best way to deal with taking appropriate consideration of the patient. This method will give the actual augmentation that is needed by the medical authorities, and the same information can be communicated distantly over the Internet. Subsequently, as a solution to traditional healthcare, there ought to be a methodology for dedicated monitoring of critical conditions for a patient. In this paper, the proposed framework will illustrate the following contributions.

- An IoT-based framework is demonstrated for the early prediction and assessment of the disease in the human body.
- A hardware module will be collecting the vital signs of a human body.
- The acquired body parameters will be transferred to the cloud via a sink node.

- Further, the real-time sensor depicted data will be monitored through a software architecture deployed in the cloud.
- The design and development of both hardware and software architecture is integrated and further evaluated for a real-time IoT healthcare application, demonstrating its promising capabilities.

The remainder of this paper is organized as follows. Section 2 discusses some of the related works. The proposed model along with the system architecture is elaborated in Sect. 3. Section 4 exemplifies the simulation environment. The detailed result and analysis are summarized in Sect. 5, and finally, Sect. 6 concludes this article with some future aspects of the proposed evolutionary framework.

2 Literature Survey

There are significant works in the literature regarding the usages of the IoT to distribute smart healthcare services. Wu et al. proposed an edge computing-based system architecture with an edge gateway along with the hybrid router for providing safety in different healthcare applications [1]. A partial integration and evaluation based upon the proposed architecture with three simulation scenarios of different IoT applications were discussed in the paper.

Taiwo et al. [4] proposed a smart healthcare support architecture for remote patient monitoring during the COVID-19 quarantine period. Different approaches have been demonstrated to identify potentially infected patients by tracing, which leads to patient isolation, and therefore, contributions are being made to slow down the spread of the virus. In another article, Singh et al. [5] described the design of an IoT-based wearable band (IoT-Q-Band) to detect the scarper. While designing the band, they considered the cost-effectiveness, global supply chain disruption, and COVID-19 quarantine guidelines recommended by the World Health Organization (WHO). They developed a mobile application along with the wearable prototype, which reports and tracks the fugitive quarantine subjects in real-time.

Nandyal et al. [6] showcased an intelligent system that had been able to monitor the current health condition of patients. The developed system could track as well as monitor patients and facilitate the management of individuals' health. Jerald et al. [7] considered authentication of IoT smart health data to ensure privacy and security of healthcare information. It illustrates different types of authentication patterns for increasing security in sensitive medical data. Greco et al. [8] led an organized review on the use of IoT in healthcare systems. This work also included a detailed discussion of the major challenges of using IoT to deliver healthcare services and classification of the swotted work in the literature.

Several other IoT-based approaches have also been anticipated by researchers in [9–13]. None of them has developed the complete framework with the help of IoT for the monitoring of body parameters. There is a need for developing and implementing a practical model which can support multiple body parameters along with
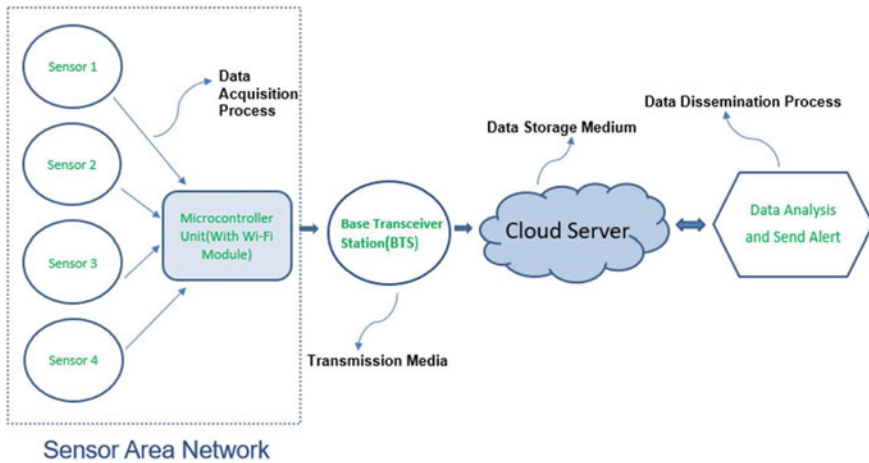
Fig. 1 Proposed model

the support for adaptive communication protocols. Compared to the existing studies, the proposed model represents a promising solution to support various healthcare applications, especially for chronic patients. It can greatly support heterogeneous sensor nodes, realizing the adaptiveness for better IoT interoperability, and Quality of Service (QoS).

3   Proposed Model

The proposed model leverages an IoT device and a Web application. The IoT device consists of wearable sensors like a temperature sensor, pulse sensor, and sink node. The sensors can able to collect the body parameters in a regular interval which is further transferred to the cloud via sink node. The sink node has an inbuilt wireless communication chip installed in it through which it can able to communicate with the server remotely [14]. First, the IoT device has to be activated to be wearied by a patient whose body vitals have to be measured. On activation, the device collects the temperature and pulses of the patient. In regular time intervals, the sensor-acquired data is sent to the server. The abnormal data are being containerized in the cloud database for further analysis and dissemination process. Various predictive models are being deployed in the cloud which helps in defining the recommendation of the doctors. In the very next step, alert message will go to the concerned medical actors for decision making.

In the proposed model, which is being depicted in Fig. 1, there are three subsections exists as follows.

Table 1 Different body parameters and its respective threshold ranges

Body parameters	Description	Threshold range
Temperature	Temperature	97–100°F
Heart rate	Frequency of cardiac cycle	60–100 BPM
ECG	Activity of heart	Frequency (0.5–100 Hz)
Respiration rate	Breathing rate	18–20 per min

- Data Acquisition
- Data Storage and Analysis
- Data Dissemination.

Following sensor nodes are being used in the data acquisition phase. Table 1 is representing the sensor nodes, characteristics and threshold values.

- Node MCU (ESP8266 Embedded Microcontroller Unit)
- DFRobot SN0203 Heart Rate Sensor
- DS18B20 Temperature Sensor.

The wearable sensor module for the data acquisition phase is shown in Fig. 2. In this phase, we have connected pulse sensor which is a well-designed plug-and-play heart rate sensor along with the temperature sensor to the human body for the collection of body parameters in regular intervals. As a temperature sensor, we have used the DS18B20 sensor along with the SN0203 pulse sensor. Further, the sensor data will go to the cloud server via a sink node. Here, we have used the Nodemcu (microcontroller) as the sink node or gateway. It has an inbuilt Wi-Fi module (ESP8266) which can transmit the data to the server via Wi-Fi.

In the data storage and analysis part, we have used mongo DB as our cloud server, where we used to store the data up to 500 MB size free of cost. As we have limited cloud storage so in every 5 min, we need to clean up the sensor data from the database
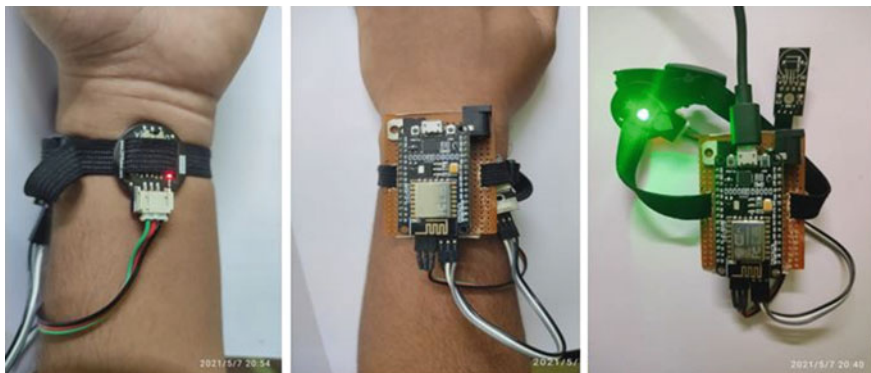
Fig. 2 Wearable sensor module
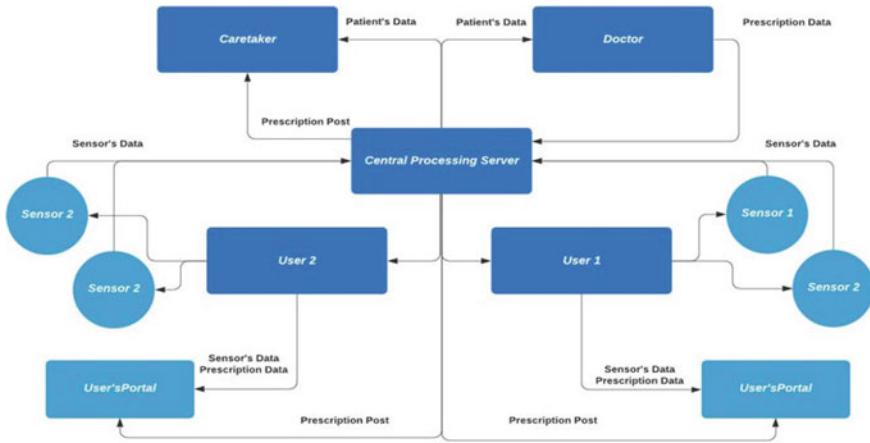
Fig. 3 Flowchart of the proposed model

for the availability of storage space for sensor data. Also, on the backend, there ought to be an information base that stores client information. It is desirable over utilize a JSON NoSQL-based information storage to use the advantages of utilizing JavaScript across the stack when similar items put away in the database can be handled by the server and the frontend with next to no extra transformation.

So, we have leveraged upon MongoDB dataset according to our necessities. Major characteristics of data storage subsection are described as follows:

- The received data from the sensor nodes is stored in a database in the cloud.
- The stored data is then logged to the patient dashboard.
- Abnormalities of the data are checked.
- Normal data is then removed from the database server after some time.

The flowchart of the model is depicted in Fig. 3. Due to the extended features like indexing, load balancing, documentation techniques, etc., we have considered the mongo database in our proposed model. Various schemas have been used throughout the simulation model for the frontend user interfacing. React is used as our primary technology stack in the frontend. It is considered for its versatile features like simplicity, data binding, native approach, testability, performance etc.

4   Simulation Environment

In the simulation model, DS18B20 is used as a temperature sensor, SN0203 as a pulse sensor and AD8232 as ECG sensor for data acquisition. The acquired sensor data from different sensors will be forwarded to the cloud server via a sink node. Nodemcu (microcontroller) is used as the sink node or gateway. It has an inbuilt Wi-Fi module (ESP8266) that can transmit the data to the cloud server via Wi-Fi.

Different sensor modules are experimented during the simulation phase [15], which are demonstrated in Fig. 4.
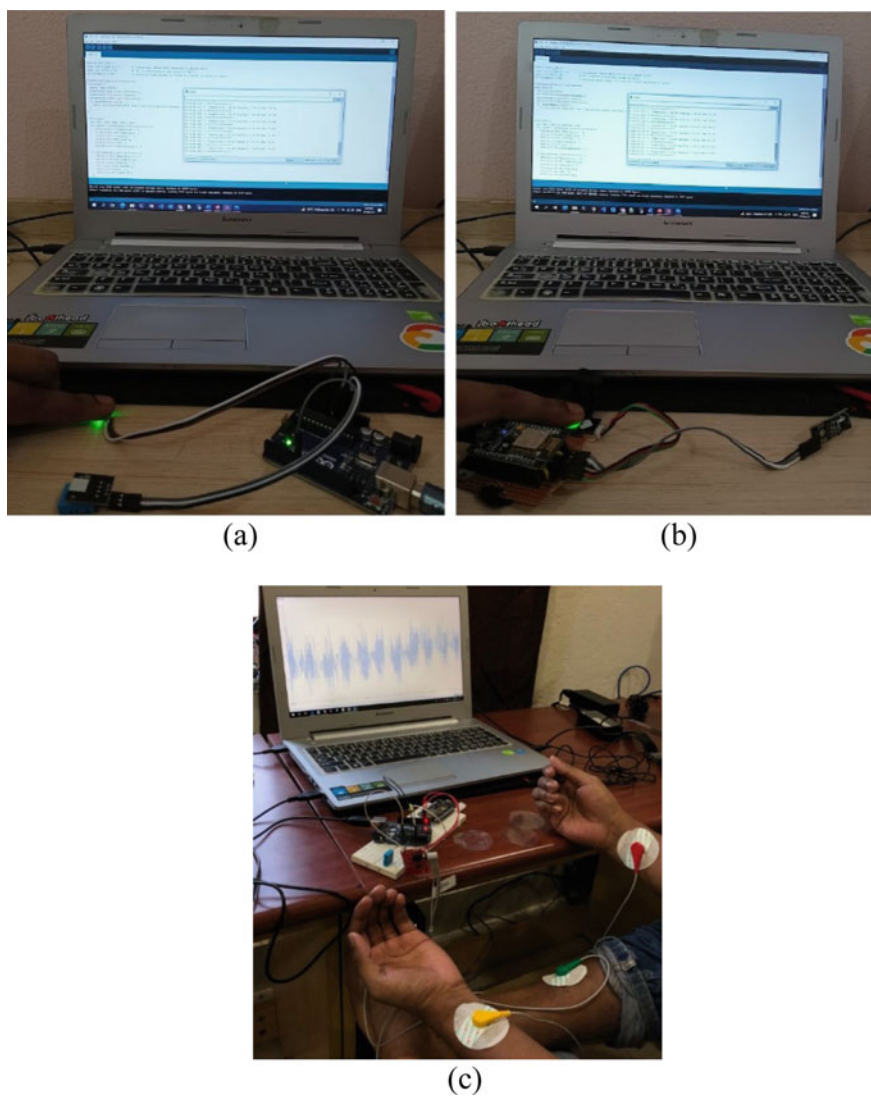
Fig. 4   Simulation environment

5 Result Discussion

A single user data has been analyzed in the cloud. A publisher–subscriber-based system is also developed in a cloud with help of React js, redux in the frontend user interface, and node js express js in the backend. Accordingly, a graphical user interface (GUI) is developed at the cloud, which is shown in Fig. 5. In the subscriber-based system, a patient gets registered to access the doctor in virtual mode. The cloud server is programmed with a threshold value for different body parameters. The body parameters are measured and depicted graphically in Figs. 6 and 7.

If any abnormalities are identified by the cloud server, then an alert message, as shown in Fig. 8, is triggered to patient, caretakers, and doctor for necessary aid. After receiving the message, doctors can log in to the portal hosted in the cloud using the credential. Doctors analyze the patient data present in the cloud and give necessary aid. The patient can log in to the portal and get detailed information provided by the concerned doctor and caretakers.

Fig. 5 Homepage of the graphical user interface

Fig. 6 Graphical representation of sensor data in cloud
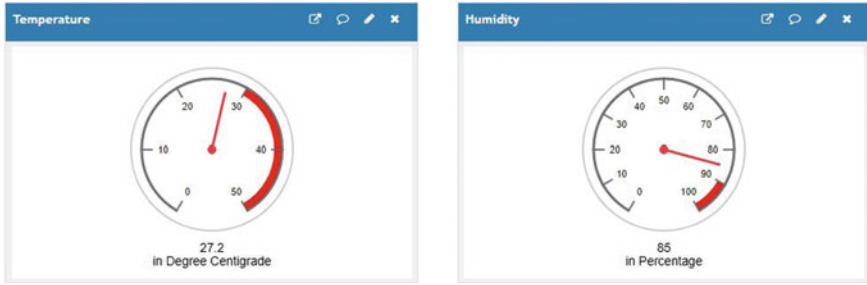
Fig. 7 Sensor values in gauge widget

Fig. 8 Alert message for
patient generated from cloud
server
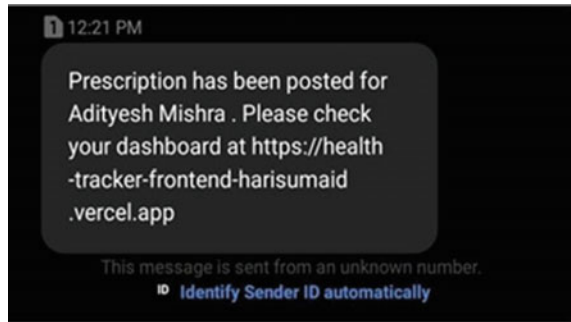
6   Conclusion and Future Scope

In the recent pandemic, situation healthcare has been treated as the most vital and prolific area for researchers. As the number of patients is increasing day by day, it is very difficult for hospitals to accommodate as well as take care of all the patients. The proposed model combines real-time sensors and the problem-solving techniques via the use of the Internet of Things. The main reason behind developing such a system is to diminish traditional healthcare overheads and also offers a system that can able to detect abnormalities instantly. The major advantage of this proposed system is to reduce the cost of the traditional healthcare system and most importantly saving the time of the patients as well as medical actors. Thus, its usage is not at all restricted or limited to any class of users. It is a very easy to handle and most efficient system, thus providing great flexibility and serving to its maximum scalability which reflects a great improvement over all other existing conventional health monitoring systems.

Remote health monitoring accompanies a ton of guarantees and has advanced into a most important help than customary medical care framework. The following trends are anticipated in the future for ubiquitous health monitoring:

- Inclusion of multiple wearable sensors for increasing the scalability aspect of the model.
- Data analysis at cloud via various deep learning and machine learning model.

- Use of adaptive sensor nodes and modern technologies with increasing information share and utilization.

References

1. Wu F, Qiu C, Wu T, Yuce MR et al (2021) Edge-based hybrid system implementation for long-range safety and healthcare IoT applications. IEEE Internet Things J 8(12):9970–9980
2. Mukati N, Namdev N, Dilip R, Hemalatha N, Dhiman V, Sahu B (2021) Healthcare assistance to COVID-19 patient using internet of things (IoT) enabled technologies. In: Materials today: proceedings
3. Gokul Krishna S, Harsheetha J, Akshaya S, Jeyabharathi D (2021) An IoT based smart outdoor parking system. In: 7th international conference on advanced computing and communication systems (ICACCS), vol 1, pp 1502–1506. IEEE
4. Taiwo O, Ezugwu AE (2020) Smart healthcare support for remote patient monitoring during covid-19 quarantine. In: Informatics in medicine unlocked, vol 20, p 100428
5. Singh V, Chandna H, Kumar A, Kumar S, Upadhyay N, Utkarsh K (2020) IoT-Q-Band: a low cost internet of things based wearable band to detect and track absconding COVID-19 quarantine subjects. In: EAI Endorsed Transactions on Internet of Things vol 6, no 21
6. Nandyal S, Gada AR (2018) A holistic approach for patient health care monitoring system through IoT. In: 2018 second international conference on green computing and internet of things (ICGCIoT), pp 68–72. IEEE
7. Jerald, Vimal A, Albert Rabara S (2020) Secured architecture for internet of things (iot) based smart healthcare. In: 2020 international conference on inventive computation technologies (ICICT), pp 828–833. IEEE
8. Greco L, Percannella G, Ritrovato P, Tortorella F, Vento M (2020) Trends in IoT based solutions for health care: moving AI to the edge. In: Pattern recognition letters, 135, pp 346–353
9. Singh RP, Javaid M, Haleem A, Suman R (2020) Internet of things (IoT) applications to fight against COVID-19 pandemic. Diabetes Metab Syndr Clin Res Rev 14(4):521–524
10. Otoom M, Otoum N, Alzubaidi MA, Etoom Y, Banihani R (2020) An IoT-based framework for early identification and monitoring of COVID-19 cases. Biomed Signal Proces Control 62:102149
11. Gope P, Gheraibia Y, Kabir S, Sikdar B (2020) A secure IoT-based modern healthcare system with fault-tolerant decision making process. IEEE J Biomed Health Inform 25(3):862–873
12. Vedaei SS, Fotovvat A, Mohebbian MR, Rahman GME, Wahid KA, Babyn P, Marateb HR, Mansourian M, Sami R (2020) COVID-SAFE: an IoT-based system for automated health monitoring and surveillance in post-pandemic life. IEEE Access 8:188538–188551
13. Nakashima N, Noda M, Ueki K, Koga T, Hayashi M, Yamazaki K, Nakagami T et al (2019) Recommended configuration for personal health records by standardized data item sets for diabetes mellitus and associated chronic diseases: a report from Collaborative Initiative by six Japanese Associations. J Diab İnvestig 10(3):868–875
14. Singh P (2018) Internet of things based health monitoring system: opportunities and challenges. Int J Adv Res Comput Sci 9(1):224–228

15. Anjari L, Budi AHS (2018) The development of smart parking system based on NodeMCU 1.0 using the internet of things. In: IOP conference series: materials science and engineering, vol 384, no 1, p 012033. IOP Publishing

A Parallel Approach to Partition-Based Frequent Pattern Mining Algorithm

Anasuya Sahoo and Rajiv Senapati

Abstract Association rule mining is one of the common way to analyze market basket data, and it provides knowledge to the decision-makers that help them to make strategic decisions. In the literature, many techniques have been studied for association rule mining but the exponential growth of data from various sources and the changing nature of data with respect to time and zone makes the analysis task trivial. In this paper, we propose a novel partition-based frequent pattern mining algorithm in order to generate robust and useful patterns from dataset in a more efficient way. The frequent patterns found from the proposed algorithm are used to generate interesting association rules. This paper provides an optimized method, which split the dataset into multiple loads on the basis of a particular attribute depending upon the number of cores available in the system, and these individual loads will get executed in parallel using our proposed algorithm. We show experimental results using datasets from retail sector to validate the capability and usefulness of our proposed algorithm.

1 Introduction

In business analytics, association rule mining (ARM) is considered as one of the most promising data mining techniques applied to discover frequent patterns and association rules. It provides actionable recommendations which are helpful in the process of decision making. But in today's world, the extensive growth of real-time data causes many difficulties in data analysis. Sometimes the rules generated from datasets were unable to convey the actual correlation among the itemsets. For example, in retail business the most precious resource for analysis is customer's data

A. Sahoo
Talentsprint, Hyderabad, Telangana, India

R. Senapati (✉)
Department Of CSE, SRM University, Andhra Pradesh, India
e-mail: rajiv.s@srmap.edu.in

and it is observed that the characteristics of these data items also change with respect to different dimensions like time and zone. Many techniques are already available in the literature to visualize the casual structure among the itemsets but most of them have given less importance on temporal data. The execution time also plays a vital role while analyzing the data from a huge repository to identify frequent patterns. One of the most popular algorithms for finding interesting pattern is apriori algorithm [1]. Another well-known frequent pattern mining (FPM) algorithm is FP growth tree algorithm. It covers up many drawbacks of traditional apriori algorithm but still it has some issues. FP tree is very expensive to construct as it follows a complex data structure. It is also not compatible with large dataset as the algorithm may not fit in the shared memory. Several algorithms were proposed for finding frequent patterns but some of them are still not suitable while analyzing large datasets. That motivate researchers to develop more advanced algorithms for FPM. The main contributions of this paper are summarized as follows.

- We propose a parallel pattern mining approach for discovering frequent patterns from a huge dataset.
- Further, we propose a novel partition-based frequent pattern mining (PFPM) algorithm to be applied in parallel using our proposed approach to discover frequent patterns.

The rest of this paper is organized as follows. In Sect. 2, we provide the review of the previous work available in the literature. In Sect. 3, we present our proposed algorithm. In Sect. 4, we discuss the experimental results. Finally in Sect. 5, we conclude this paper.

2 Related Work

Business analytics include the statistical understanding of data for effective communication and to develop problem-solving techniques. For better understanding of transnational dataset, many algorithms like Apriori, FP-growth, and E-clat were proposed. These algorithms mine the dataset and produce frequent patterns to generate rules. In FPM for finding interesting patterns, traditional apriori algorithm is one of the very well-known algorithms proposed in [1]. But multiple dataset scan, computational time, consumption of memory are the major drawbacks of this algorithm which motivated researcher to develop new hybrid algorithms. The work reported in [2] explained a novel method to generate both positive and negative association rules and also defined the impact of negative association rule which will help in decision making. In [3], an improved T_Apriori algorithm has been proposed which reduces the computational time by 98% in average. It performs much better with the increment of the transaction numbers and minimum support value. An improved apriori algorithm is proposed in [4], which performs single database scan for frequent pattern generation. In [5], an improved apriori algorithm is proposed which reduces 67.38% of the time consumption. A new count-based method is proposed

to prune candidate itemsets and used to generate rules to reduce total number of dataset scan in [6]. In [7], pruning optimization and transaction reduction based on improved apriori algorithm is proposed which enhanced the computational time. In [8], an improved apriori algorithm is proposed for mining regular itemset using bit matrix which need only single scan of the transnational dataset to build compressed data structure. In [9], apriori hybrid algorithm is presented by merging the features of both weighted apriori and hash-based apriori algorithms for semi-structured data. In [10], apriori-growth algorithm is proposed that includes the advantages of apriori algorithm and the FP tree structure for handling a huge dataset is an effective manner. A boolean load matrix and advanced binary matrix-based frequent pattern mining algorithm are proposed in [11–13], respectively. Both of the approaches scan the dataset only once in order to find out all the frequent itemsets.

From this detailed survey, we found that a wide range of work has been already done in this area. However, the increasing volume of data and analyzing these huge amount of data to produce convenient rules with minimum execution time is still a big challenge. Further, in the distributed environment, we also need to maintain certain criteria like load balancing and maintaining minimum inter-process communication. Hence, in this paper, we have introduced a PFPM algorithm that satisfies all these requirements by optimizing the frequent pattern generation technique within a distributed environment.

3   Working Principles

In this paper, we have provided a parallel pattern mining approach for finding the frequent patterns in an efficient manner. We have distributed the dataset into multiple non-overlapping partitions on the basis of a particular characteristic of the data, and this technique will help us to minimize inter-partition communication. We are considering partitions having equal number of attributes in order to maintain load balancing. Then, the association rules are generated from frequent itemsets which satisfies the minimum confidence threshold. The detailed architecture of our proposed approach is presented in Fig. 1. Here the transaction dataset is distributed into multiple load matrices on the basis of a specific attribute, and each load is then executed by our proposed algorithm. Further, distributed association rule mining (DARM) is used to generate association rules. The entire task is performed in parallel using multiple cores.

3.1   Partition-Based Frequent Pattern Mining Algorithm

Here we have proposed a novel algorithm that takes three inputs, and these are transaction dataset denoted by D_t, minimum support denoted by α_{min}, and number of cores denoted by x. The transaction dataset D_t includes a set of items, i.e., $(I_0, I_1, ..., I_n)$
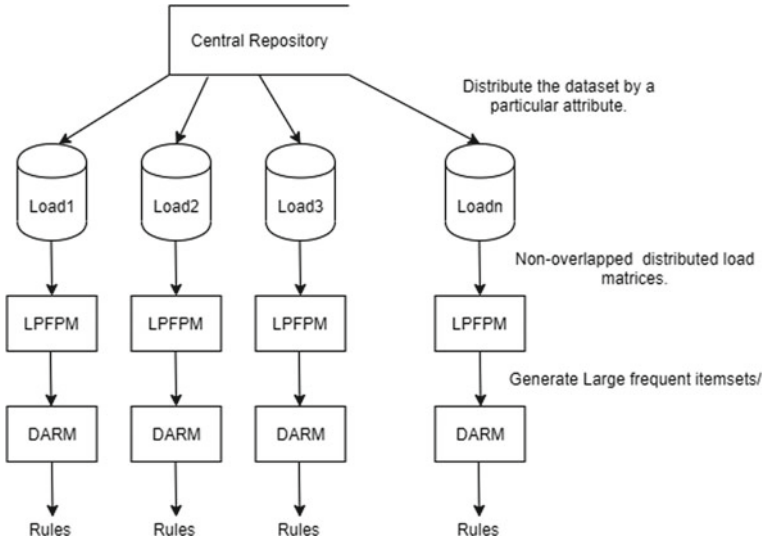
Fig. 1 Architecture of parallel pattern mining approach

and transactions, i.e., $(T_0, T_1, ..., T_m)$. Further, it is converted into boolean matrix where items and transactions are stored in rows and columns, respectively. The matrix is presented in Eq. (1).

$$
\begin{bmatrix}
a_{00} & a_{01} & \ldots & a_{0n} \\
a_{11} & a_{12} & \ldots & a_{1n} \\
a_{21} & a_{22} & \ldots & a_{2n} \\
& \ldots \ldots \ldots & & \\
a_{m1} & a_{m2} & \ldots & a_{mn}
\end{bmatrix}
\tag{1}
$$

Each a_{ij} is presented by boolean constants that presented in Eq. (2).

$$
a_{ij} = \begin{cases} \text{true} & \text{if } T_j \in I_i \\ \text{false} & \text{if } T_j \notin I_i \end{cases}
\tag{2}
$$

The support count value for candidate-1 itemset, i.e., (C_1), is calculated by taking the summation of all the transactions for a particular item. The calculated support values are compared with minimum support, and the results are stored in L_1. The L_1 matrix is vertically divided into x partitions, i.e., $partition-1, partition-2, ..., partition x$, which is presented in Eq. (3).

$$
\text{partition}_s[i][j] = L_1[i][j], i \leq n, j \leq t, t = \frac{m}{x}, 1 \leq s \leq x
\tag{3}
$$

Further, $AND(\&)$ operation is performed on these partition matrices to calculate the support count values for each candidate-p itemset which is presented in Eq. (4).

$$sup.r = \sum_{r=1}^{x} \sum_{i,j=1}^{m,t} \text{partition}_r[i][j] \ \& \ \text{partition}_{r+1}[i][j] \qquad (4)$$

If the $sup.r$ value is greater than minimum support, then that particular itemset is added to frequent-p itemset, i.e., L_p. The large frequent itemset, i.e., L, can be computed by taking the union of all frequent itemsets, i.e., $L_1, L_2, ...L_{p-1}, L_p$. The procedure to discover the frequent patterns from a transaction dataset is explained through algorithms in multiple phases. The proposed Algorithm 1 takes the dataset, i.e., D_t, minimum support, i.e., $\alpha_{\min}$, and the total number of cores x as an input and produces large frequent itemset, i.e., L as an output. The detailed algorithm is executed as follows:

Algorithm 1 PFPM Algorithm

Input: D_t , α_{min} , x
Output: L
1: Construct $Q_{m \times n}$ from D_t
2: For $i \leftarrow 0$ to m
    For $j \leftarrow 0$ to n
    Calculate $\alpha_{sup}[i] = \alpha_{sup}[i] + Q[i][j]$
    if $(\alpha_{sup}[i] \geq \alpha_{min})$ then
    $L_1 = L_1 \cup Q[i :]$
    end if
    end For
    end For
3: Set $div = m/x$
4: For $i \leftarrow 1$ to x
    For $j \leftarrow 0$ to n
    For $k \leftarrow 0$ to div
    $partition_i[j][k] = L_1[j][k]$
    end For
    end For
    end For
5: C_p=**Generate_candidate**(L_{p-1}), $p \geq 2$
6: For each $i, i \in C_p, p \geq 2$
    γ_{sup}=**Compute_sup**($C_p[i], i$)
    if $(\gamma_{sup} \geq \alpha_{min})$ then
    $L_p = L_p \cup C_p[i]$
    end if
    End For
7: Repeat step-5 and step-6 until $L_{p-1} = \Phi$
8: $L = \bigcup_{i=1}^{p} L_i$

- In step 1, the dataset D_t is scanned to construct the boolean matrix $Q_{m \times n}$.
- In step 2, for each i and j, $i < m$, and $j < n$, the $Q[i][j]$ values of individual columns are added and stored in $\alpha_{sup}[\ ]$. If any individual value of α_{sup} is greater than equal to the pre-assumed α_{min}, then the entire tuple is added to frequent-1, i.e., L_1 matrix.
- In step 3, div is assigned with m/x value where x is the total number of selected cores.
- In step 4, for each i, $1 \le i \le x$, for each j and k where $0 \le j \le m$ and $0 \le k \le$ div, the values of frequent-1 matrix, i.e., $L_1[j][k]$ is distributed and stored into individual partitions, i.e., partition$_i$.
- In step 5, Generate_candidate() is called to perform the join operation over frequent-$(p-1)$ itemset, i.e., L_{p-1} where $p \ge 2$. It returns candidate-p itemset, i.e., C_p as output.
- In step 6, for each itemset $i \in C_p$, Compute_sup() is called. It takes $C_p[i]$ as argument where $i \in C_p$, $p \ge 2$ and returns support count of the itemset, i.e., γ_{sup}. If the $\gamma_{sup}[i]$ is greater than equal to α_{min}, then the itemset $C_p[i]$ is added to frequent-p itemset, i.e., L_p.
- Steps 5 and 6 are repeated until $L_{p-1} = \Phi$.
- In step 8, all the frequent itemsets are combined to generate large frequent itemset, i.e., L.

Algorithm 2 Generate_candidate() Algorithm

Input: L_{p-1}
Output: C_p
1: Let $q_1, q_2 \in L_{p-1}$
2: Set $m = |L_{p-1}|$
3: For each $i, i \le y$
   For each $j, j \le y$
   if $q_1[i] \ne q_2[j]$ then
   $C_p = C_p \cup \{q_1[i] \cup q_2[j]\}$
   end if
   end For
   end For
4: **Remove_item(C_p)**
5: return C_p

In Algorithm 1, Generate_candidate() method is called in step 5 to perform join operation between the itemset with itself. Algorithm 2 is taking the previous frequent-$(p-1)$ itemset, i.e., L_{p-1} as input parameter to produce the next candidate-p itemset, i.e., C_p. The detailed algorithm is explained below:

- In step 1, we have considered two frequent itemsets q_1 and q_2 such that both $q_1, q_2 \in L_{p-1}$.
- In step 2, m contains the total number of individual itemset present in L_{p-1}.

- In step 3, for each i and j (i.e., $i \leq y$, $j \leq y$), $q_1[i]$ join with $q_2[j]$ only if both itemsets are distinct (i.e., $q_1 \cap q_2 = \phi$). The itemset, which satisfies the condition, is added to C_p.
- In step 4, Remove_item() method is called to perform the prune operation.
- In step 5, the updated candidate-p itemset (i.e., C_p) is returned.

Remove_item() method is used to delete the itemsets whose subsets are not a part of the previous frequent itemset. In Algorithm 3, candidate-p itemset (i.e., C_p) is given as input parameter. The algorithm is executed as follows:

Algorithm 3 Remove_item() Algorithm

Input: C_p
Output: C_p
1: Let $d \in C_p$
2: For each (p-1) subset d
   if $d \notin L_{p-1}$ then
   delete d
   end if
   end For
3: return C_p

- In step 1, we have considered d to represent each $(p - 1)$ itemset of C_p.
- In step 2, for each subset $d, d \in C_p$ is deleted if p is not present in frequent-$(p - 1)$ itemset (i.e., L_{p-1}).
- In step 3, the pruned candidate-p itemset (i.e., C_p) is returned to Algorithm 2.

In Algorithm 1, Compute_sup() method is called in step 6 to calculate the support count (i.e., γ_{sup}) from individual partitions.

Algorithm 4 Compute_sup() Algorithm

Input: C_p
Output: sup_r
1: Set $l = 1$
2: Set $\gamma_{sup} = 0$
3: For each $r, r \leq x$
   For each $j, l \leq j \leq (l + m/x)$
   $\gamma_{sup} = \gamma_{sup} + partition_r[i][j] \& partition_{r+1}[i][j]$
   end For
   Set $l = j$
   end For
4: return γ_{sup}

- In step 1, l is assigned with 1.
- In step 2, initially the support count (i.e., γ_{sup}) of a particular itemset i is taken as 0 .
- In step 3, for each r, $r \in x$ and for each j where $j \in [l, l + m/x]$, AND operation is performed between each element of partition$_r$ and partition$_{(r+1)}$ and the result is added to γ_{sup}. Then l is assigned with the value of j.
- In step 4, finally γ_{sup} is returned to the calling method.

After the execution of the PFPM algorithm, we have found all frequent itemsets (i.e., L) which satisfied the minimum support threshold. Here the dataset is scanned only once while generating frequent-1 itemset which helps to reduce the complexity in terms of computational time.

4   Result and Discussion

In this section, the proposed algorithm is applied to a real-world problem. We have explained the importance of PFPM algorithm which generates a large number of patterns and rules. Mostly, we have focused on how these rules can be more meaningful and helpful in a good decision-making process. Using the parameters mentioned in Table 1, we have implemented PFPM algorithm for finding frequent patterns. We have distributed the dataset over *date of purchase* attribute where the Load-1, Load-2, and Load-3 represent summer, rainy, and winter data, respectively. Each load will get executed in individual cores. Again, in the PFPM algorithm the frequent-1 matrix was vertically divided into multiple partitions like partition-1, partition-2, and partition-3 depending on the number of cores that have been chosen. These partitions are needed to be mutually exclusive in order to avoid inter-partition communication and must have equal number of columns for load balancing purpose. The DARM technique is applied on the output of PFPM algorithm to find out the interesting rules. After successful execution of PFPM algorithm, there were 3121, 3062, 2893 rules generated by Load-1, Load-2, and Load-3 datasets, respectively, whereas only 35 association

Table 1 System parameter

System parameter	Value
Dataset	Grocery-shop.csv
Processor	Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz (8 CPUs), 1.8GHz
Memory	8192MB RAM
Minimum Support	0.0003, 0.0005, 0.0007, 0.0009, 0.0011
Minimum confidence	0.05
Mimimum length	3
Load − wise distribution	Seasons
x	3

(a) Number of rules generated   (b) Top 15 Frequent items
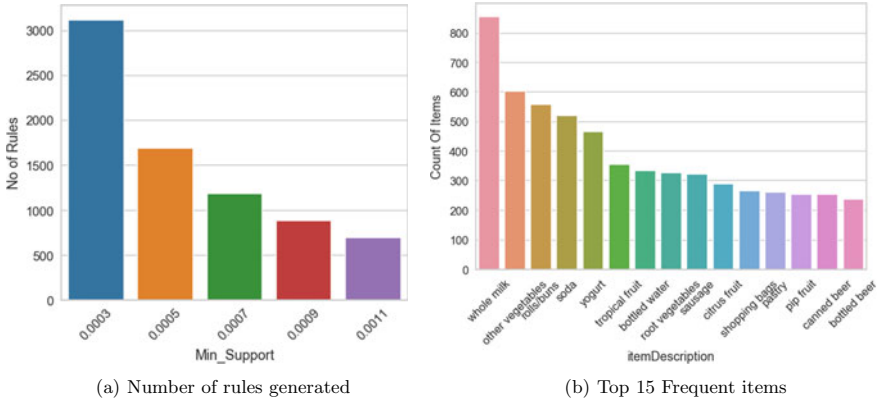
Fig. 2 Number of rules generated and frequent items in summer.

rules were generated by the traditional apriori algorithm from grocery shop dataset while considering the minimum support as 0.0003, minimum confidence as 0.05, and minimum length as 3.

It indicates that our proposed algorithm generates more no of patterns which will help in analyzing the itemsets. Further, to observe the relationship between the min_support value with number of rules generated, we have calculated the number of rules generated with different min_support values for the entire dataset which is shown in Fig. 2a. The graphs clearly indicate that irrespective of seasons, the number of association rules gradually decreases with the increasing value of min_support. Hence, we have to choose the min_support value considering the total number of transactions present in the dataset, which will help to analyze the purchase rate of any item pairs. Figure 2b represents the most frequently brought 15 items with respect to summer season. It plots the items against their support count values. We observed that few items, and their frequencies both were changing with respect to seasons and few items were redundant irrespective of seasons. Hence, it is very important to analyze the pattern of the items which will help to make strategic decisions for a profitable business.

Here we have compared the execution time of the proposed PFPM algorithm with traditional apriori algorithm. Using the parameters described in Table 1, we have executed PFPM algorithm in two different ways, i.e., with complete grocery-shop.csv dataset and with load-wise distributed dataset with different support value. We have found that the overall performance of PFPM algorithm with and without load-wise distributed dataset is 94.92 and 84.68% better than existing apriori algorithm respectively. The major reason behind such performance is the single scan of given dataset. In each pass, all the calculations are performed from its previous pass. After the load division, the calculations are performed in parallel in different cores.

5   Conclusion

Mining frequent pattern from a large dataset is an important research task. An efficient solution to this task brings lots of insights into the data for its better usage while making important decision. To conclude, in this paper, we introduce a novel partition-based frequent pattern mining algorithm in order to generate robust and useful patterns in a more efficient way. Firstly, we have distributed the dataset into non-overlapping partitions on the basis of a characteristic of the dataset. Secondly, we have applied our proposed PFPM algorithm to generate frequent patterns from it. Then the frequent patterns are applied to generate association rules with support, confidence, and lift values. We have considered an exemplary dataset from retail sector to illustrate the working principle and usefulness of the proposed algorithm and also showed the experimental results to validate the feasibility and effectiveness of our proposed algorithm. Further, the proposed algorithms can be suitably modified and applied in a distributed environment using Hadoop framework.

References

1. Agrawal R, Imieliński T, Swami A (1993) Mining association rules between sets of items in large databases. In: ACM SIGMOD conference, Washington DC, USA, vol 22(2), pp 207–216
2. Sim ATH, Indrawan M, Srinivasan B (2008) The importance of negative associations and the discovery of association rule pairs. Int J Bus Intell Data Mining 3(2):158–176
3. Yuan X (2017) An improved apriori algorithm for mining association rules. In: AIP conference proceedings, vol 1820(1), pp 1–6
4. Binhua L (2009) An improved algorithm of apriori. In: Springer's international symposium on intelligence computation and applications, pp 427–432
5. Al-Maolegi M, Arkok B (2014) An improved apriori algorithm for association rules. arXiv preprint arXiv:1403.3948, pp 427–432
6. Wu H, Lu Z, Lin P, Xu R, Wenbao J (2009) An improved apriori-based algorithm for association rules mining. In: IEEE International conference on fuzzy systems and knowledge discovery, vol 2, pp 51–55
7. Chen Z, Cai S, Song Q, Zhu C (2011) An improved apriori algorithm based on pruning optimization and transaction reduction. In: IEEE International conference on artificial intelligence, management science and electronic commerce, pp 1908–1911
8. Dutt S, Choudhary N, Singh D (2014) An improved apriori algorithm based on matrix data structure. Global J Comput Sci Technol
9. Thakre KR, Shende R (2017) Implementation on an approach for mining of datasets using APRIORI hybrid algorithm. In: IEEE international conference on trends in electronics and informatics, pp 939–943
10. Wu B, Defu Z, Qihua L, Jiemin Z (2008) An efficient frequent patterns mining algorithm based on apriori algorithm and the FP-tree structure. In: IEEE International conference on convergence and hybrid information technology, vol 1, pp 1099–1102
11. Sahoo A, Senapati R (2020) A Boolean load-matrix based frequent pattern mining algorithm. In: IEEE's International conference on artificial intelligence and signal processing (AISP)
12. Patro PP, Senapati R (2021) Advanced binary matrix-based frequent pattern mining algorithm. In: Intelligent systems. Springer, pp 305–316
13. Sahoo A, Senapati R (2021) A novel approach for distributed frequent pattern mining algorithm using load-matrix. In: International conference on intelligent technologies (CONIT), pp 1–5

A Novel Task Offloading and Resource Allocation Scheme for Mist-Assisted Cloud Computing Environment

Subhranshu Sekhar Tripathy, Kaushik Mishra, Rabindra K. Barik, and Diptendu S. Roy

Abstract Nowadays, the demand for Internet of Things is increasing rapidly throughout the world, thanks to the speed with which wide area networks may be easily reached by numerous users. Due to the increasing growth of the Internet of Things, certain computation will be necessary for significant calculations, such as cloud computing and edge computing, which are now more secure for storing IoT-related data. To manage the massive amount of data collected from various IoT gadgets, this paper proposes a unique approach for offloading jobs to a mist-assisted cloud environment in order to reduce the overall expected schedule and enable very rapid response to diverse IoT events. This method makes use of "fuzzy logic algorithms," which take into consideration application attributes, resource deployment, and assortment. As mist-assisted cloud environments are built from the bottom-up, they consist of four layers: IoT devices, mist level, fog level, and cloud level (service provider). The mist controller is a centralized module that is used to manage the scheduling, placement, and control of applications in a mist-assisted cloud environment. The mist controller is composed of four components: an application supervisor, a support manager, a control unit, and a developer. The mist controller is a self-contained equipment that is responsible for scheduling offloading duties in order to fulfill the requirements of mist cloud application users and system requirements. In a series of simulated simulations, the suggested approach is compared to previous approaches that have been found to improve total service time for latency-sensitive applications and make effective use of mist-assisted cloud resources. Additionally,

S. S. Tripathy (✉) · D. S. Roy
Department of Computer Science and Engineering, National Institute of Technology, Meghalaya, India
e-mail: P19CS013@nitm.ac.in

D. S. Roy
e-mail: diptendu.sr@nitm.ac.in

K. Mishra
Department of Computer Science and Engineering, Sambalpur University Institute of Information Technology (SUIIT), Burla, India
e-mail: kmishra@suiit.ac.in

R. K. Barik
School of Computer Applications, KIIT Deemed to be University, Bhubaneswar, India

the findings indicate that, depending on the computing resources and communication types available, different decommissioning options inside the mist-assisted cloud environment may result in a varying time period.

Keywords Fog computing · Cloud computing · Mist computing · Load balancing · Resource utilization

1 Introduction

In current IT world, more than 50 billion IOT devices are connected. It provides a stable high-speed computation environment for IOT application. As fog computing expands the current cloud services to the edge of n/w, it is required to decide where application should execute and their service requirements can be supported [1]. For that an efficient load balancing task, scheduling approach is required. Selecting customized real-time task for executing at fog layer improves overall task execution efficiency of cloud and IOT application. The development of ICT-based solution meets the requirements of resource utilization [2]. In order to manage the quality-of-service demand, context-aware computing, that is, fog computing is implemented. An intelligent sleep–awake mechanism is used in fog layer for effective allocation of resources.

Resources allocation and task scheduling are the main feature of data center which other way manage the load balancing. Main aim of resources allocation is to optimize number of active physical machine and work balance [3]. Compared to cloud computing, fog devices has limited power supply. Computation resources and communication resources and design challenge to meet real-time retirement. For resource balancing and management, multiple fog applications are running for energy-efficient offloading decision mechanism. Fog computing resolves the latency and mobility support problem. The working principle of our brain is more similar to the form in which fuzzy logic operates. The main objective of fuzzy logic is to minimize the complexity of a problem to a manageable level [4]. The methods of fuzzy logic can easily adjust to the changing nature of computational resources and application parameters while also enabling scalability within the system's context [5].

The primary contributions of this research are enlisted as follows:

- Fuzzy logic is used to offload the IoT request to different layers based on the priority of tasks,
- A task scheduling algorithm is implemented to map the tasks on to suitable fog nodes,
- Four-layered architecture is used to offload the users' request to reduce the service rate and response time.

The following are the features of the present article: Sect. 2 explores related literature and provides a complete overview of "cloud, fog, mist, and IoT computing" in

a collaborative setting. The proposed model for load balancing and optimal resource utilization approaches is explained in Sect. 3. Following a discussion, Experimental outcome and simulation analysis result have been depicted in Sect. 4. Finally, Sect. 5 concludes with concluding comments and recommendations for the future.

2 Related Work

2.1 Mist-Fog-Cloud Architecture

Cloud computing is characterized as a collection of thousands of computers connected with varying several functions and documentations, simplified in the cloud with the hubs which are inextricably linked via Internet. Parallel and distributed computing architectures are also implemented additionally by the above method. Fog processing is the type of network planning which can be used as Internet of Things gadgets to perform calculations, accumulate, and exchange data in precise region [6, 7]. "Fog computing" is a collaborative environment in which IoT applications are deployed. The computing environment faces various challenges like structural design, interface with programming, get rid of computing, sharing of resource, load balancing, data security along with storing of data that has been examined in [1]. Throughout the data transfer between IoT-enabled devices and fog servers, processing latency and transmission latency will be affected [8]. Mist computing allows computer hardware to be placed on the edge of an organization's original devices. Microchips or microcontrollers connected to physical objects will process all information detected by objects. After finding this information, it will pass the desired information on to the organization [9]. For long-term data, mist computing demands less cloud storage and transport power. The mist processing environment, for example, runs adjacent to physical devices, providing a low-power gateway that boosts throughput while lowering latency at the device's edge [10].

2.2 Resource Utilization Strategies

Organizing offloading assignments depending on resource usage or resource assortment has been accepted substantial significant interest as of several researchers. In the context of cloud computing, resource allocation plays a very important role [11]. If the available resources are not efficiently allocated between different Web servers, then it becomes a crucial moment. The profit of using dynamic resource allocation in a computing environment is lower localization costs and no software or hardware overhead. Previously, researchers have illustrated that offloading is not a new technique as it is used in cloud computing. To increase mobile application execution performance and overall energy economy, offloading moves calculations

from resource-limited mobile devices to resource-rich cloud nodes [2]. It is a simple way to empower IoT apps by outsourcing heavy processing duties to the mist-aided cloud environment's powerful servers [3]. Furthermore, it is utilized to overcome the limits of IoT devices in terms of computational power (e.g., CPU and memory) and battery life. This is one of the most essential IoT enabling strategies since it permits a sophisticated computing activity to be performed beyond the capabilities of the device [12]. Several academics have expressed an interest in scheduling offloading tasks depending on resource utilization or resource heterogeneity.

2.3   *Load Balancing*

It is the process of distributing requests among all the servers. It is carried out to assure that the system is not over-burdened. The merits of load balancing techniques are numerous. They can help improve the system's efficiency and performance by reducing the number of steps and waiting for a response time reduction [13]. The mist controller is a traffic inspection scheme between server and client demands. The goal of a balancer is to enhance the exploitation of available resources. The available resources must be distributed in a viable way. The advantages of dynamically allocating resources are numerous, especially when compared to the overhead and hardware constraints of traditional computing environments [20–22]. In addition, the use of load balancing techniques helps minimize server downtime. Each server has its own unique set of connections, and the information about it is updated continuously [3]. The load balancer sends the query for the server in sequence. A dynamical round-robin method is similar to the round-robin in that it assigns the weight to the servers according to the load along with capacity of server.

3   Proposed Supply Efficient Mist-Assisted Model

As revealed from Fig. 1, the mist-assisted cloud environments worn bottom-up approach which comprises of four layers such as IoT devices, mist level, fog level, and cloud level (service provider). The IoT layer consists of groups of connected devices (e.g., smart phones, autonomous cars, and intelligent video surveillance); these devices have a variety of functions in which each function has multiple [14]. The above-mentioned facilities can be installed performed on a variety of computing resources (mist node, fog node, or cloud) where the support manager and application manager must choose where to apply the above-noted facilities [15]. Here, in this planned work, each mist computing node is placed at the border levels that are nearer to the IoT devices at the base station [9]. These nodes were dispersed geographically and could belong to the same cloud provider or other brokers [16]. Every edge node has a node manager which can manage compute resources and start application services [17]. All border nodes are connected to the mist controller. Mist controller
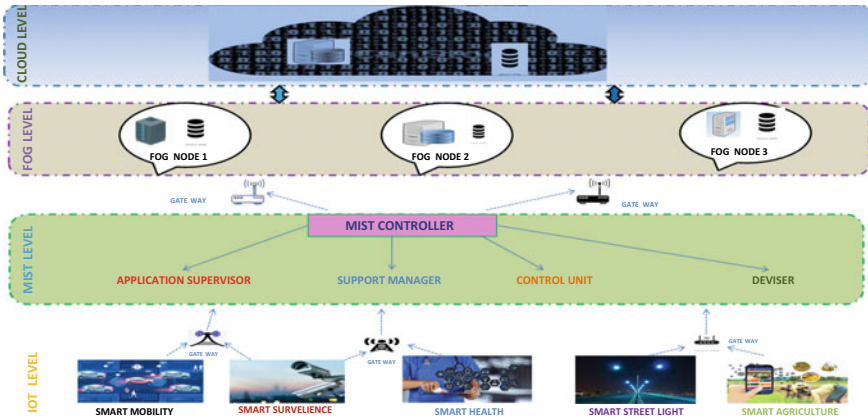
Fig. 1 Proposed supply efficient mist-assisted cloud environment

is a centralized module that is used for scheduling, placement, and controlling applications facilities in the mist-assisted cloud environment. To get the detailed position of resources in architecture (e.g., accessible and worn), amount of Internet of Things devices, its application's task, and at which place tasks have been allocated (mist, fog, or cloud) the mist controller communicates with all its components [18].

There are four parts to mist controller: application supervisor, support manager and designer. Mist controller is a separate unit that schedules offloading tasks to satisfy user and system needs. This mist layer organizes the jobs assigned to the edge nodes. It also ensures data synchronization with the cloud. It manages the duties involved in running apps in the mist-assisted cloud architecture. This list usually includes data reassignment requirements, calculation requirements, and delay constraints. The support manager is in charge of all physical assets linked to the mist-aided cloud framework. It includes CPUs, networking, and IoT devices. This component helps the mist-aided cloud framework to ignore application tasks like computational and communication latency. It will also monitor CPU usage while running other apps. The control unit of the mist-aided cloud framework will also resolve any shortage of computing resources or job execution failure owing to network difficulties. Deviser completes the mist controller. In the mist-aided cloud environment, this component recommends a scheduling method for offloading jobs and the location where they will be located.

The deviser component proposes the task offloading mechanism and reports its findings to the mist controller for implementation. The proposed approach will gather information on offloading workloads and server utilization. Mandatory inputs include edge VM use, task length, data transmitted per job, and delay sensitivity. All of these variables are linguistic variables. The three levels are low, medium, and high. The VM utilization parameter indicates the current level of VM utilization on the local mist and fog server. Task length measures the task's computing load in MIPS (MIPS). This value represents the network demand for uploading and downloading jobs.

Making an offloading decision requires determining where to transfer the task. The task's delay sensitivity is a property that assesses how sensitive it is to latency. During the fuzzification phase, the "fuzzifier" will receive mathematical input from system infrastructure monitoring and arriving processes. Then, assign each value to its linguistic variable (e.g., low, medium, high) [19]. The method of transforming the outcome of the fuzzy rules to a particular significance according to the o/p membership function is recognized as defuzzification. There are many methods for producing the o/p membership function in a fuzzy logic system, and some of them are most popular ones. In order to determine the best target layer for offloading the computation tasks, Algorithm 1 describes in detail.

Algorithm 1 presents the task offloading algorithm using fuzzy logic. Based on the fuzzy logic architecture, the task offloading decisions are devised. Here, the tasks are offloaded into three different layers based on the priority of the task. Fuzzy logic system takes the fuzzy input variables and processes them to produce the desired target layer. This algorithm primarily focuses on mapping of tasks to compatible fog nodes or cloud VMs.

Algorithm 1: Task offloading algorithm using Fuzzy logic

INPUT: IoT request (Ti) with the parameters $T_{len}, T_{network}$ and T_{delay}

OUTPUT: Choose the compatible resources at the destination tire R_{mist}, R_{cloud} and R_{fog}

Step 1 for 1 to n

Step 2 Calculate average VM utilization V_{avg}

Step 3 f = Fuzzification($T_i^{len}, T_i^{network}, T_i^{delay}, V_{avg}$)

Step 4 If $f <= f_{low}$ fless then

Step 5 Assign Ti on R_{mist}

Step 6 else

Step 7 If f $<= f_{med}$ then

Step 8 Assign Ti on R_{fog}

Step 9 else

Step 10 Assign Ti on R_{cloud}

The number of connected Internet of Things devices, task duration, and number of cycles required to complete the task and task aim are all collected from the start. Algorithm 2 depicts the entire process of allocating each calculation work to a processing resource. This is how computational assignments are made. To begin, the edge node's applications, workloads, and computational resources are listed. As shown in Algorithm 2, line 1, the most difficult tasks arrive first, followed by the easiest. The edge node's available computing resources are also ordered by central processing unit (CPU) power, with the most powerful resources displaying first. The procedure then repeats over the efficient jobs, assigning each to the appropriate computer resources, with the heavier duties going to the influential available computing resources as indicated in Algorithm 2.

Algorithm 2 shows the task scheduling algorithm. Here, the list of tasks is sorted based on the descending order of their task length, whereas all the fog resources are arranged in the descending order based on their capacity. If the task requirement is

less than or equal to the corresponding resource capacity, then the respective task is allocated to the compatible fog resource.

Algorithm 2: Task Scheduling Algorithm

INPUT: Set of IoT devices Di, Set of tasks T_j and Set of resources at Fog node F_{re}

OUTPUT: Mapping of Task T_{ij} to compatible Fog resources F_{re}"

Step 1 Arrange the task in the task list in descending order based on their task length

Step 2 Arrange the fog resources in the resource list R in descending order based on their capacity

Step 3 For 1 to D_n

Step 4 While T_{ij} is not mapped

Step 5 if $T_{ij}^{CPU} \leq R_c^{CPU}$ then

Step 6 Allocate T_{ij} on R_c

Step 7 Increment T_{ij}

Step 8 else

Step 9 Increment R_c

---

4 Result and Analysis

The proposed method was tested in a mist cloud environment and compared to other comparable research using an "Edge-Cloud Sim" simulation program and utilizing mist-assisted cloud resources to improve inclusive service time and job collapse for "latency-sensitive applications." All trials were simulated on Edge-CloudSim to validate our proposed mist assisted cloud infrastructure. Newly exploited computations in mist registering for "Utilization-Based, Sonmez and Flores" are distinct and proposed "load-balancing algorithm." With this strategy, resource utilization and work offloading are greatly improved. We tested the algorithm's efficiency using dynamic autonomous tasks of various length. Figure 2 illustrates the average service time for the four techniques vs. the number of gadgets, with service time consisting of both processing and network time. The studies were designed to improve resource management in a mist-assisted cloud environment after decreasing latency for Internet of Things applications. When the systems are unloaded, performance of every process is essentially identical, as seen in the figure. However, when the number of IoT devices grows, the proposed approach's service time remains constant in comparison to the other alternatives.

The proposed algorithm is generic and scalable to handle the peak loads of the IoT devices. Due to the implemented fuzzy logic, the tasks are categorized in to different groups and forwarded to different layers for the processing. An improved task scheduling algorithm is implemented to allocate the task on to the compatible fog nodes efficiently. Therefore, the considered conflicting scheduling parameters are improved significantly.
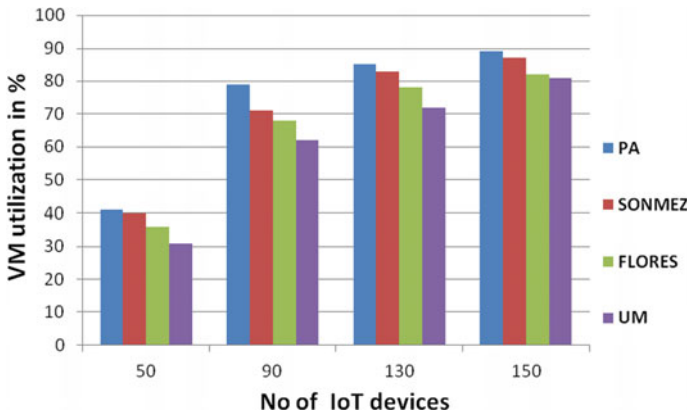
Fig. 2 Resource utilization of the PA with other related approach

5   Conclusions and Future Scope

In this research, a new task offloading strategy for latency-sensitive IoT applications in mist-assisted cloud environment is introduced and assessed. To significantly reduce service time and maximize resource utilization, this approach needs to take into account application aspects (e.g., central processing unit request, communication request along with delay sensitivity), as well as the flexibility and assortment of resources. It also took into account various forms of computational resources that depict a realistic scenario. To assess the suggested approach, various techniques for scheduling offloading jobs are simulated. The findings reveal offloading task scheduling techniques that ignore application aspects and system operation that might cause service time dilapidation for latency-sensitive functions. Furthermore, in terms of overall service time and resource consumption, the proposed approach outperforms other relevant approaches when it comes to task offloading. It can help cut down on overall job failures caused by network and computational resource difficulties.

References

1. Roy DS, Behera RK, Reddy KHK, Buyya R (2019) A context-aware fog enabled scheme for real-time cross-vertical IoT applications. IEEE Internet Things J 6(2):2400–2412
2. Shang C, Chang C-Y, Liu J, Zhao S, Roy DS (2020) FIID: feature-based implicit irregularity detection using unsupervised learning from IoT data for homecare of elderly. IEEE Internet Things J 7(2):10884–10896
3. Mishra K, Majhi S (2020) A state-of-art on cloud load balancing algorithms. Int J Comput Digit Syst 9(2):201–220
4. Tripathy SS, Roy DS, Barik RK (2021) M2FBalancer: a mist-assisted fog computing-based load balancing strategy for smart cities 13(3):219–233

5. Reddy KHK, Luhach AK, Pradhan B, Dash JK, Roy DS (2020) A genetic algorithm for energy efficient fog layer resource management in context-aware smart cities. Sustain Cities Soc 63:102428

6. Barik RK, Dubey H, Samaddar AB, Gupta RD, Ray PK (2016) FogGIS: fog computing for geospatial big data analytics. In: IEEE international conference on electrical, computer and electronics engineering (UPCON), pp 613–618

7. Clarke A, Steele R (2011) How personal fitness data can be re-used by smart cities. In: IEEE Seventh International Conference on Intelligent Sensors, Sensor Networks and Information Processing, pp 395–400

8. Reddy KHK, Behera RK, Chakrabarty A, Roy DS (2020) A service delay minimization scheme for QoS-constrained. Context-Aware Unified IoT Appl IEEE Internet Things J 7(10):10527–10534

9. Liu Y, Fieldsend JE, Min G (2017) A framework of fog computing: architecture, challenges, and optimization. IEEE Access 5:25445–25454

10. Phi NX, Hung TC (2017) Load balancing algorithm to improve response time on cloud computing. Int J Cloud Comput: Serv Archit 7(6):1–12

11. Narczyk P, Siwiec K, Pleskacz WA (2016) Precision human body temperature measurement based on thermistor sensor. In: IEEE 19th International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS), pp 1–5

12. Hussain M, Beg MM (2019) Fog computing for internet of things (IoT)-aided smart grid architectures. Big Data Cogn Comput 3(1):12–23

13. Yuxuan J, Huang Z, Tsang DHK (2018) Challenges and solutions in fog computing orchestration. IEEE Netw 32(3):122–129

14. Tarik T, Samdanis K, Mada B, Flinck H, Dutta S, Sabella D (2017) On multi-access edge computing: a survey of the emerging 5G network edge cloud architecture and orchestration. IEEE Commun Surv Tutorial 19(3):1657–1681

15. Almutairi J, Mohammad A (2021) A novel approach for IoT tasks offloading in edge-cloud environments. J Cloud Comput 10(1):1–19

16. Scoca V, Atakan A, Ivona B, Rocco DN, Rafael BU (2018) Scheduling latency-sensitive applications in edge computing. In: Closer, pp 158–168

17. Choi N, Kim D, Lee S-J, Yi Y (2017) A fog operating system for user-oriented iot services: Challenges and research directions. IEEE Commun Mag 55(8):44–51

18. Imagane K, Kenji K, Jiro K, Toshitaka T, Hidenori N (2018) Performance evaluations of multimedia service function chaining in edge clouds. In: 15th IEEE annual consumer communications & networking conference (CCNC), pp 1–4

19. Sonmez C, Ozgovde A, Ersoy C (2019) Fuzzy workload orchestration for edge computing. IEEE Trans Netw Serv Manage 16(2):769–782

20. Abdollah A, Bakar A (2014) A comparative study of three artificial intelligence techniques: genetic algorithm, neural network, and fuzzy logic, on scheduling problem. In: IEEE 4th international conference on artificial intelligence with applications in engineering and technology, pp 31–36

21. Tripathy SS, Barik RK, Roy S (2022) Diptendu: "Secure-M2FBalancer: a secure mist to fog computing-based distributed load balancing framework for smart city application. In: Advances in communication, devices and networking, Springer, Singapore, pp 277–285

22. Barik RK, Misra C, Lenka RK, Dubey H, Mankodiya K (2019) Hybrid mist-cloud systems for large scale geospatial big data analytics and processing: opportunities and challenges. Arabian J Geosci 12(2):1–32

Analysis of Performance Characteristics in Social Wireless Sensor Network

Madhusmita Patra and Sasmita Acharya

Abstract In Wireless Sensor Networks (WSNs), the sensor nodes are interconnected and communicate with each other wirelessly to collect data about the surrounding environment. The nodes sometimes selectively hide some information, and selectively expose some data, generate and forward data the network and connect/disconnects to the network accordingly. Ideas from social networks have been employed on the basis of the assumption that the nodes in the network have their specific social life, to show how the nodes communicate in the social network to achieve significant efficiency. As wireless sensor networks monitor specific environments, the data flow toward fixed sinks from the sensors. With growth of Internet of Things (IoT), the services need the data to be sent directly to operator-specified devices. Then, the data sets off immediate actions for the services at the device. This service is known as a device-to-device interactive IoT service. In this paper, WSN has been studied as a social senor network in which the embedded sensor nodes are the main entities as opposed to human as in a traditional social network. Also the paper discusses on the architecture of a Social WSN and compares its performance with that of other contemporary models.

1 Introduction

"Wireless Sensor Networks (WSNs)" is a network of sensors arranged either inside a specific region that needs to be sensed. The sensor node consists of components responsible for sensing, data processing, and communication. Architecture of the WSN is in general fixed in nature and the sensed data flow toward predefined gateway. The gateway analyzes the overall data. After the monitored information

M. Patra (✉) · S. Acharya
Department of Computer Application, VSSUT, Burla, India
e-mail: madhupatra90@gmail.com

has been gathered, the action to be performed next is determined with respect to the consolidated data.

After the evolution of "Internet of Things (IoT)," a number of services have been provided recently. For example, for a smart building and CCTV. The environment that is monitored by sensors is real time. The collected data is sent to the required actuators. The devices carry out immediate actions according to the information, conditions and series of operator and relationship with respect to the service. This service is called as "device-to device interactive IoT service." The WSN can cause significant delay because of delay caused for data collection and need of human determination. The determination by human or server is processed in a local domain.

There are number of characteristics about the service. The sensor and devices should be sociable for each service being provided. A fire sensor event should have sociability with only fire appliances to prevent a disaster. The intruder sensor should have sociability with security device. But movement-sensing information has a number of socialites with other devices that depends on the numerous services. The service, being independent of each other, consists of specific job-related sensors and relates devices. The selected sensors and devices have a service-specific role. They transfer the information among themselves with respect to the relationship defined by the operator. A domain may have multiple independent services the same time. Finally, we can design and deploy the service that is flexible and dynamic with respect to the service environment and requirement of operators.

The sociability should be applied dynamically and flexibly over WSNs for supporting multiple independent "device-to-device interactive IoT" services. According to the sociability of the services, the relationship between sensed data and its flow should be able to configure WSNs in a dynamic manner; and the sensors and devices should be flexible.

In this paper, "Social Wireless Sensor Network (SWSN)" is studied where the design and procedures provide sociability on the top of WSNs. In Social WSN, needed sensors and devices are selected, and the role of the sensor is configured i.e., source, sink. It establishes the data flow in between the sensors and devices with respect to the sociability of given service dynamically. And hence, it can come up with auto-configurable WSN for every "device-to-device interactive IoT" service.

The organization of the paper is as given. Section 2 discusses related work done in WSN. Section 3 represents the architecture of the Social Wireless Sensor Networks along with the description of its layers and functionalities. Section 4 provides the results of simulation, and Sect. 5 concludes the paper.

2   Related Work

The researcher has proposed the architecture, functions of "Social Wireless Sensor Network," and the sociability of the services over wireless sensor networks in [1]. In [2], the author has surveyed different application area of "sensor networks" that has been proposed. In [3], the "Internet of Thing (IoT)," enabling factor of this

paradigm, integration of several technologies and communications solutions has been addressed. Article [4] provides vision and challenges to realize an IoT. The article [5] reviews some challenges and promises of CPS, followed by unification of some specific challenges and promises that are Sensor Networks. In [6], a possible architecture of IoT has been proposed that provides navigation of social network, and characteristics of the IoT network structure has been analyzed. The [7] proposes possible solution to issues related to things' service discovery and composition that is to be addressed in "Internet of Things." In [8], "Cyber Physical System" platforms and systems that have been developed recently is reviewed. The [9] describes the abstraction of virtual sensors for enabling an application developer to programmatically decide high-level data requirements for an application. In [10], infer-virtual information sensing methods have been reviewed. [11] proposes "Neuro-Fuzzy Optimization Model (NFOM)" to facilitate fault-tolerance WSNs. In [12], PEGASIS protocol has been studied. [13] describes the directed diffusion protocol. In [14], hierarchical routing protocols have been studied including TEEN.

3 Overview of Social Wireless Sensor Networks

This section represents the framework of a "Social Wireless Sensor Network" (SWSN). The SWSN can be classified into three layers:

- Physical layer
- Social layer
- Service layer.

The sensors, things, and gateway provide some extra functions at the layers to facilitate sociability for "device-to-device interactive IoT" assistance. The thing should be an active device like smart phones, CCTV etc. (Fig. 1).

Physical Layer: The physical layer consists of data collecting sensors (both sink and source). The gateway takes care of the overall WSN. The network is self-organized by the sensors and fixes the behavior as either source or sink. The data are delivered to the sinks in a data-centric manner. It delivers the control information for configuring SWSNs in a dynamical manner with respect to service.

Social Layer: The social layer delivers service dedicated SWSN. The thing's position determines the participating sensor in the Social WSN. The thing's functionalities decides the role of the sensors and the path of goal-centric data transmission in SWSN. The behavior of sensor is flexible both as source and sink; path for data flow is set up in dynamical manner with respect to the sensors depending on the sociability profile of things in every service.

Service Layer: The service comprises of service-related "active and passive things." The organic relationship, that is, sociability defines a service. The service defines the things, its role and data delivery correlation among the things for the service
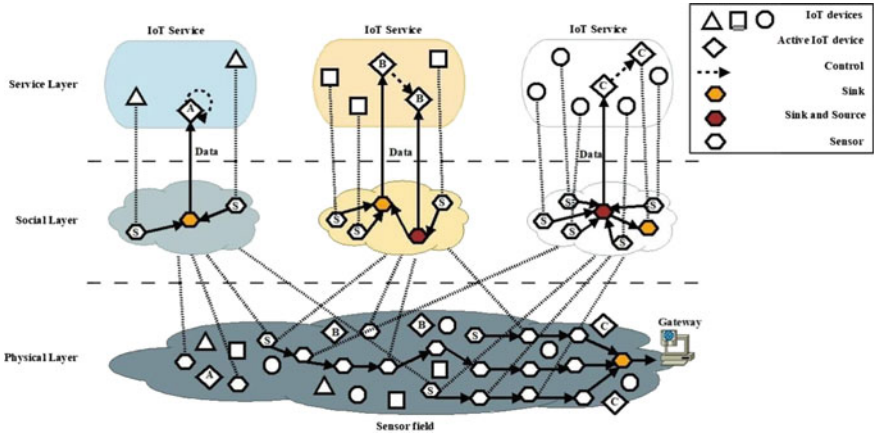
Fig. 1 Network framework of SWSN [1]

objective. Business logics are carried out by the things according to the Social WSN data.

3.1 System Functions

Things, gateway, and sensors execute system roles at the three layers as shown in Fig. 2.

Configuration Network of SWSN: Traditional self-organized WSN is utilized as the configuration network.

Sociabi1lity Profile: Sociability profile contains the service ID, participating things ID, the thing's role and the path for data dispatch between things.

Dynamic Service Subscription: WSNs discover the services by the "dynamic service subscription" of things dynamically.

Auto-Selection of Social Sensors: The configuration network chooses the participating sensors for SWSN in the time of service subscription delivery to the sink. Social sensor is the sensor(s) on the way of Configuration network from source to sink in SWSN.

Binding Social Grades: The social sensors generate social grades with respect to the sociability profile.

Dynamic Configuration of a SWSN: Social WSN is formed when a thing connects to service, and Social WSN topology is changed when more connect to the service; topology changes if thing(s) go away or withdraw, and Social WSN is removed when all things withdraw from the network formed.
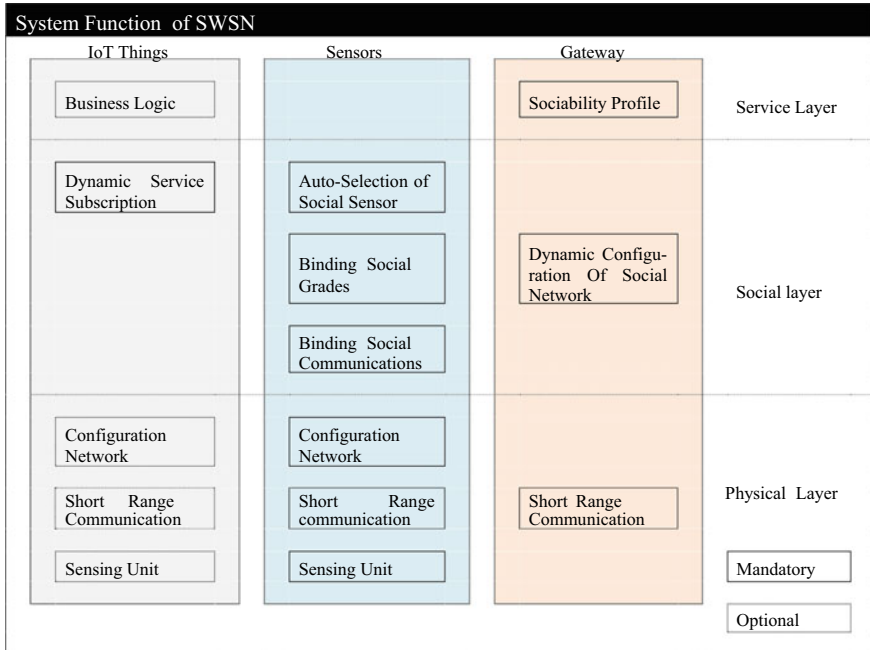
Fig. 2 System functions of SWSN

Binding Social Communication: A "device-to-device interactive IoT" services need communication procedures for the efficiency of the service. As physical WSN generally supports single communication procedure in network, and the Social WSN should maintain transparency of transmission between the social and physical layer for every service.

4   Results of Simulation

This part presents simulation results and their analysis. Here, the Social WSN performance has been evaluated via simulation and compared with that of "Directed Diffusion" [13], "TEEN," [14] and "PEGASIS" [12] which are some well-known routing protocols in WSNs. The simulations have been done in MATLAB R2014a.

Table 1 represents the values for different simulation criteria.

The following results are based on the existing model which was discussed in 2nd section. In this section, we have compared the existing model with some other WSN protocols.

Figure 3a shows the length of the path with respect to the end-to-end distance. It is observed that the path length of SWSN is 6.92 hops for end-to-end distance 70 m; but the path length of the directed diffusion is 23.40 hops. The path length for

Table 1 Simulation criteria

Parameters	Value
Size of network	100 m * 100 m
No. of nodes	400
No. of rounds	400
Packet size	30 B

TEEN and PEGASIS routing protocols lies in between that of SWSN and Directed Diffusion.

Figure 3b shows the delay impacted by the end-to-end distance. In order to support the interactive service, delay should be shorter. It was observed that the delay in SWSN was much lower for the end-to-end distance 70 m than that of "Directed Diffusion," "TEEN," and "PEGASIS".

Figure 3c shows the network lifetime with reference to the number of simulation rounds. It was observed that the SWSN has a comparatively higher network lifetime than a typical WSN implementing Directed Diffusion, TEEN, and PEGASIS.

Figure 3d shows the residual energy with respect to the number of simulation rounds. It was observed that the SWSN has a higher network lifetime than that of a typical WSN implementing Directed Diffusion, TEEN, and PEGASIS protocol.

5   Conclusion

The SWSN applies the sociability dynamically over traditional WSNs. The Social WSN functions sociability, subscription of a service, auto-selection of social sensors. The social grades decide the flow of data among the social sensing devices. The social communication binding provides a well-organized communication. Social WSN framework supports multiple numbers of IoT services over WSNs. It also provides auto-configurable custom designed service according to its sociability. The simulation results validated that the performance of SWSN was comparatively much better than that of other contemporary routing protocols for WSNs like Directed Diffusion, TEEN, and PEGASIS. This work may be further extended to facilitate the expansion of customer-centric IoT services so as to cater the needs of the present scenario.
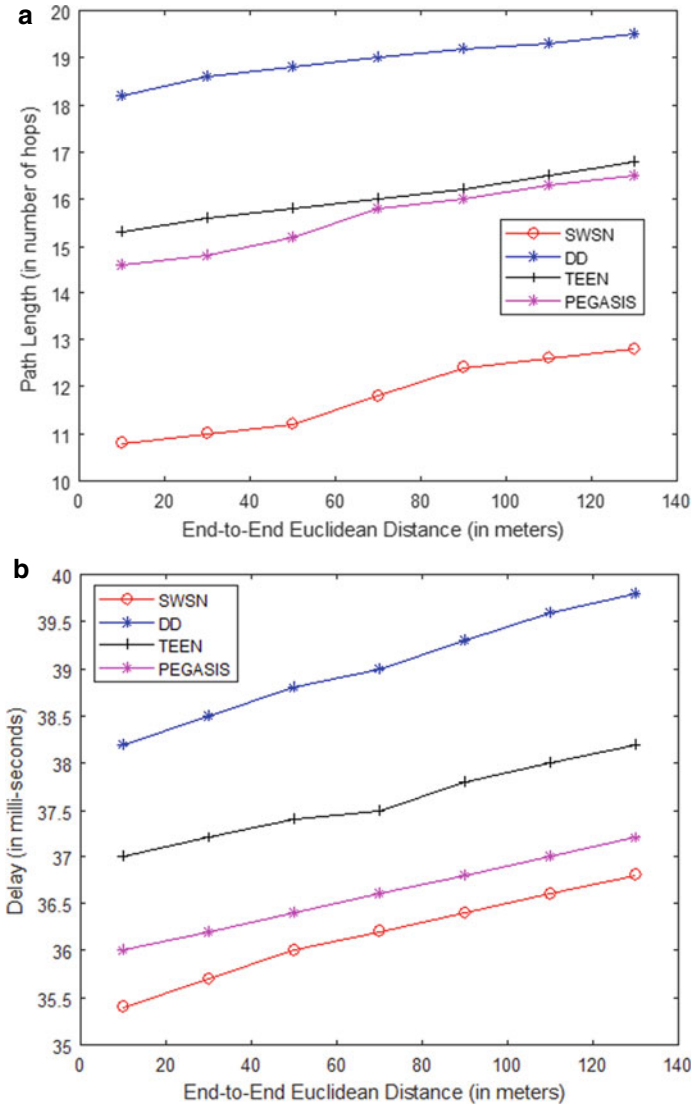
Fig. 3 **a** Length of Path due to End-to-End Distance. **b** Delay affected by End-to-End Distance. **c** Lifetime of the network with reference to by number of Rounds. **d** Residual Energy with respect to number of Rounds
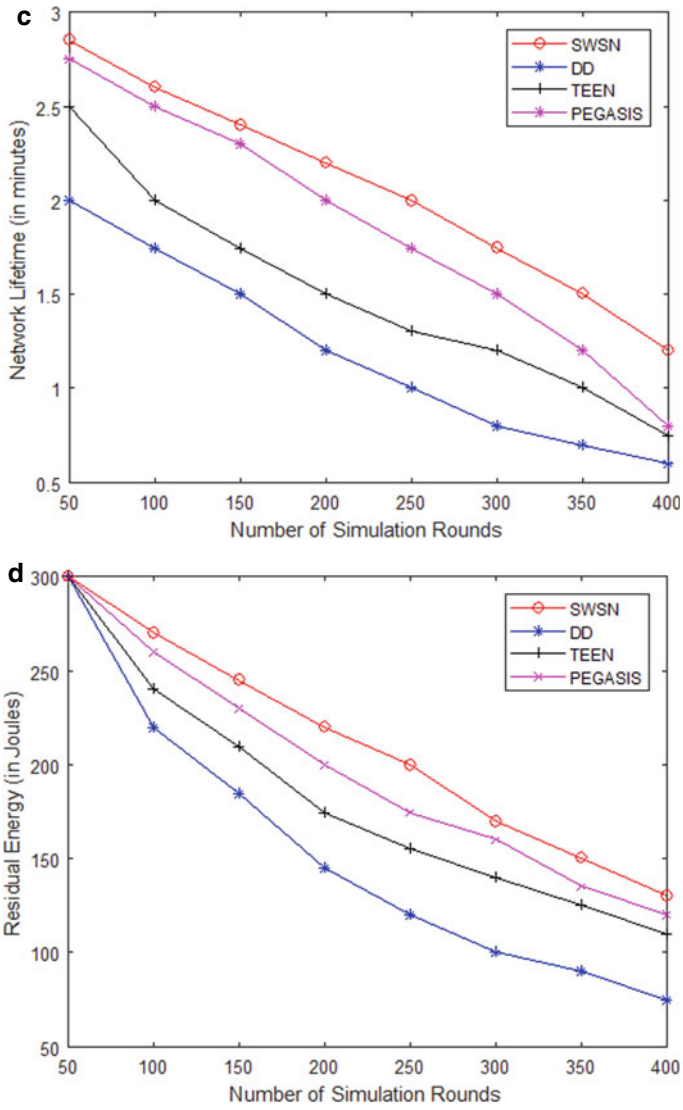
Fig. 3 (continued)

References

1. Kim S, Yim YY (2016) Social wireless sensor network toward device-to-device Interactive Internet of Things Services. Int J Distrib Sens Netw 12(9)
2. Arampatzis Th, Lygeros J, Manesis S (2005) A survey of applications of wireless sensors and wireless sensor networks. In: Proceedings of the 13th Mediterranean conference on control and automation, Cyprus, pp 719–724, 27–29 June 2005
3. Atzori L, Iera A, Morabito G (2010) The Internet of Things: a survey 54(15):2787–2805
4. Sundmaeker H, Guillemin P, Friess P (2010) Vision and challenges for realizing the Internet of Things. Office of the European Union, Luxembourg
5. Sha L, Gopalakrishnan S, Liu X (2008) Cyber-physical systems: a new frontier. In: Proceedings of IEEE international conference on sensor networks, ubiquitous, and trustworthy computing, Taichung, Taiwan, pp 1–9, 11–13 June 2008
6. Atzori L, Iera A, Morabito G (2012) The Social Internet of Things (SIoT)—when social networks meet the Internet of Things: concept, architecture and network characterization. 56(16):3594–3608
7. Atzori L, Iera A, Morabito G (2011) SIoT: giving a social structure to the Internet of Things 15(11):1193–1195
8. Wu FJ, Kao YF, Tseng YC (2011) From wireless sensor networks towards cyber physical systems. Pervas Mob Comput 7(4):397–413
9. Kabadayi S, Pridgen A, Julien C (2006) Virtual sensors: abstracting data from physical sensors. In: 2006 international symposium on a world of wireless, mobile and multimedia networks, NY, 26–29 June 2006
10. Liu L, Kuo SM, Zhou M (2009) Virtual sensing techniques and their applications. In: International conference on net- working, sensing and control (ICNSC'09), Japan, pp 31–36
11. Acharya S, Tripathy CR (2018) An ANFIS estimator based data aggregation scheme for fault tolerant wireless sensor networks. J King Saud Univ Comput Inform Sci 30:334–348
12. Rana H, Vhatkar S, Atique M (2014) Comparative study of PEGASIS protocols in wireless sensor network. IOSR J Comput Eng (IOSR-JCE), 2278–8727, 16(5):25–30
13. Rawat U, Sharma M (2012) Directed diffusion: features, current developments, issues and analysis. Int J Comput Appl 49(12):975–8887
14. Xu DW, Gao J (2011) Comparison study to hierarchical routing protocols in wireless sensor networks. In: 3rd international conference on environmental science and information application technology, pp 595–600

Finite Element Analysis of an Optimized Sandwiched Spur Gear Set

Sumanta Panda, Jawaz Alam, and Arup Pradhan

Abstract Complex shape and geometry of gear lead to consideration of a large number of design parameters in the gear design. Traditional design approaches have the challenge of determining design variables that meet only a single design criteria at a time, which is insufficient for high performance and dependability. This design approach meets the demand for compact, efficient, and reliable spur gear by reducing the computational complexity and cost. In this article, Particle Swarm Optimization (PSO) algorithm is used for optimizing the parameters of the single-stage spur gear set. Three different materials i.e., structural steel, EN24, and Hylum are sandwiched together in a different arrangement. Stress analysis has been performed using ANSYS. The deformation of the gear teeth has been analyzed. The design optimization, content in terms of methodology, and gear material selection for composite spur gear set presented in this paper are encouraging enough to attract researchers working in this field.

Keywords Spur gear · Sandwich-type gear set · Contact stress · Deformation · FEA

1 Introduction

One of the main aims of optimum spur gear set design is to guarantee a long-term design of spur gear sets with minimum contact stress at various points of contact, which extends fatigue life. Lightweight spur gears have a wide range of applications in industry, and they've also been investigated as a possible machine element in power transmission systems in aviation and space technologies. To determine the maximal stress and stress fields on gear tooth flanks during tooth meshing, Stanojevic and Cvejic [1] introduced the approach. The contact of gear teeth flanks was modeled using FEM. Based on the results of equivalent stress distribution in the contact zone, the authors have proposed that FEM is one of the superior numerical approaches for

S. Panda (✉) · J. Alam · A. Pradhan
Department of Mechanical Engineering, VSSUT, Burla, Odisha 768018, India
e-mail: spanda_me@vssut.ac.in

contact stress analysis for the tooth flank of the mating gear. In another study, Hassan [2] has developed a program to estimate the contact stress on the contacting teeth. To generate ten cases, this program was executed for each 30 rotations of the pinion from the initial point contact to the endpoint contact. This program provided the graphic analysis for the gear teeth of different FE models and the analysis of stress was performed in ANSYS. The obtained outcomes from the FEA were compared with the theoretical calculations. The standard gear set without linear tip relief has been meshed and studied by Markovic et al. [3] using the FEM to illustrate the comparisons between the contact stress value obtained in FEM and the estimated value of contact stress from Hertz's theory (theoretical estimation). Gupta et al. [4] have performed the FEA of gear with the help of ANSYS 13 to evaluate Hertizan's stress generated between two mating spur gears and compared the results with theoretical Hertzian's equation. Based on the estimated results, the authors suggested that, hardness of the tooth profile can be enhanced to oppose surface fatigue i.e., pitting. Also, the contact stress of the gear teeth decreases with an increase in the module. Also, if contact stress reduction is the prime focus and if the high torque has to be transmitted then a spur gear with a larger module is favored. Rani et al. [5] have adopted nylon, polycarbonate, and cast iron as the spur gear materials for FEA. The authors have suggested that the deflection of the cast iron is higher than others materials i.e., nylon and polycarbonate. As a result, cast iron gears may be replaced with nylon to achieve high efficiency, long life, and low noise. Mahebub et al. [6] have performed a comparative study using FEA on non-metallic and metallic spur gears. The 3D model of spur gear was developed using Pro-engineer modeling software and simulation is performed in ANSYS workbench and the authors have reported that non-metallic gears provide more benefit simply for a limited strength application compared to metallic gears. Sajad et al. [7] analyzed the dynamic response of gear using FEA. The gears were operated at different rotating speeds and the corresponding stress distributions, displacements, and strains were estimated. The authors have reported that maximum stress occurs at the tooth root and bending stress rises with the rise in the speed of the gear. Sachidananda et al. [8] highlighted one of the major methods for reducing contact stresses or Hertizan's stress in spur gear sets by using addendum shift with altered tooth sum technique. SEM images of the gear are studied to investigate the morphology of gear wear. Al-Qrimli et al. [9] have performed a static FEA on 3D models of the spur gears made up of steel in the numerical method with the help of ABAQUS software. The contact stress values obtained in FEA are compared with the contact stress value determined by using Hertz's theory (analytical approach) for validation purposes. The authors reported that the magnitude of maximum contact stress occurs at the pitch circle of the gear tooth. In a recent study, Alam et al. [10, 11] have reported an multi-objective optimization approach to reduce the contact stress or Hertizan's stress along the path of contact by using profile modification with altered tooth sum approach. In their study, the authors demonstrated profile modification as a useful approach to reduce contact stress and the weight of the gear set. Furthermore, Alam and Panda [12] have extended the work of [10, 11] and adopted PSO-TLBO and Jaya algorithm to solve this multi-objective optimization
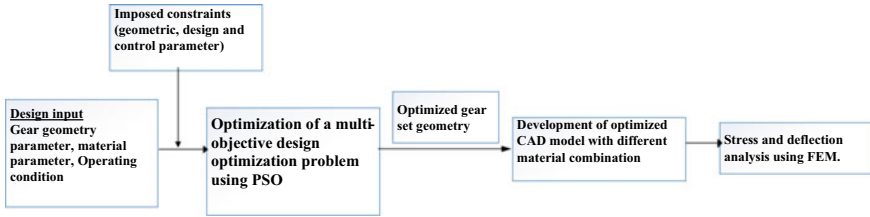
Fig. 1 A block diagram for implementation of the proposed method

problem. Jaya algorithm is more efficient than PSO and PSO-TLBO algorithm in terms of cost function [12].

The extensive literature review reveals that very limited literature related to the minimization of contact stress at different points of contact for composite spur gear set has been explored. The literature shows that a sandwich-type gear set has not been most popularly considered for stress analysis. However, it seems a much interesting approach that gives enough strength to the gear set to withstand high load and may reduce the weight of the gear set. The design analysis considered both metallic (Structural steel and En24) and composite materials (Hylum) material and sandwich together. Further, the FEA has been performed which compares the maximum stress induced in the gear sets with a different material combination. It may help the reader in selecting an appropriate gear set arrangement with a material combination as per the operational requirements. An elaborate detail on this aspect is presented in the result and discussion section. The rest of this manuscript is structured as follows. The methodology is discussed in Section. 2. Section 3 presents the result and discussion. Finally, the study is concluded in Section. 4. The implementation steps of the design optimization have been presented in Fig. 1, which gives an idea to the researchers about the methodology and analysis of gear design adopted in this study.

2   Methodology

In this paper, we have studied the contact stress of a different spur gear set i.e., set 1, set 2, set 3. Further, the gear set has been developed by fixing the layer of gear sets of different materials at alternate combinations (as can be seen in Table 1). The optimum gear dimension has been obtained by the meta-heuristic approach i.e., PSO. Afterward, a CAD model has been developed to check the geometrical feasibility, and stress analysis has been performed in ANSYS. A basic design of single-stage spur gear is presented in Fig. 2 [13]. The important parameters of the gear set are the same as in [13].

Table 1 Face width (mm) distribution of gear set

Face width	Layer 1	Layer 2	Layer 3
Set 1	7.108	7.108	7.108
Set 2	8	5.32	8
Set 3	9	3.32	9
Comb.1	Hylum	EN24	Structural steel
Comb.2	EN24	Structural steel	Hylum
Comb.3	EN24	Hylum	Structural steel
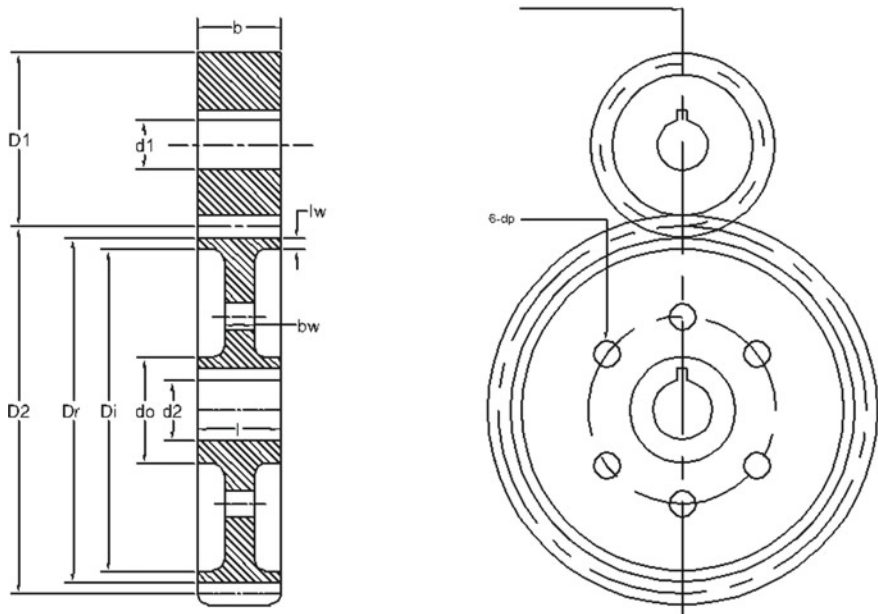
Fig. 2 Spur gear set

2.1 Design Variable

In this article, an attempt has been done to assert if any enhancement in the outcomes is feasible by applying PSO algorithms to the constraint optimization problem. The optimum design problem involves 10 non-linear constraints as reported in [13] and 5 design variables (i.e., b, d_1, d_2, H, and x_1). The input design parameters with their ranges are reported in Table 2.

Table 2 Input design parameters

Design variable	Lower bound	Upper bound	Variable types
b (mm)	10	35	Continuous
d_1 (mm)	10	20	Continuous
d_2 (mm)	10	40	Continuous
H (BHN)	200	400	Continuous
x_1	0.8	1.5	Continuous

2.2 PSO Algorithm

Kennedy and Eberhart [14] first demonstrated PSO, a population-based optimization technique. At each stage of the PSO, the velocity (or acceleration) of particle is changed toward its 'p Best' and 'g Best' locations (global version of PSO). A random term is used to weight the acceleration, with different random numbers created for acceleration toward the 'p Best' and 'g Best' locations. The particles' velocity and position are updated according to Eqs. (1) and (2). The detailed steps of the PSO algorithm are explained in [14]. Table 3 presents the optimum dimensions obtained from the PSO algorithm.

$$V_{i+1}[] = V_i[] + C_1 * \text{rand}() * (p_{\text{Best}}[] - \text{Current}[])$$
$$+ C_2 * \text{rand}() * (g_{\text{Best}}[] - \text{Current}[]) \tag{1}$$

$$\text{Current}_{i+1}[] = \text{Current}[] + V_{i+1}[] \tag{2}$$

Table 3 Optimum dimensions of gear set

Parameter	Value	Parameter	Value
B	21.33	b_w	7
m	2	D_1	20
z_1	18	D_2	36.4
z_2	72	d_o	61.4
a	90	d_p	16.9
R_{11}	18	I_w	5
R_{12}	72	D_r	139
Ra_1	20	D_i	129
Ra_2	74	D	95.2
R_{d1}	15.5	R_{b1}	16.31
R_{d2}	69.5	R_{b2}	65.25
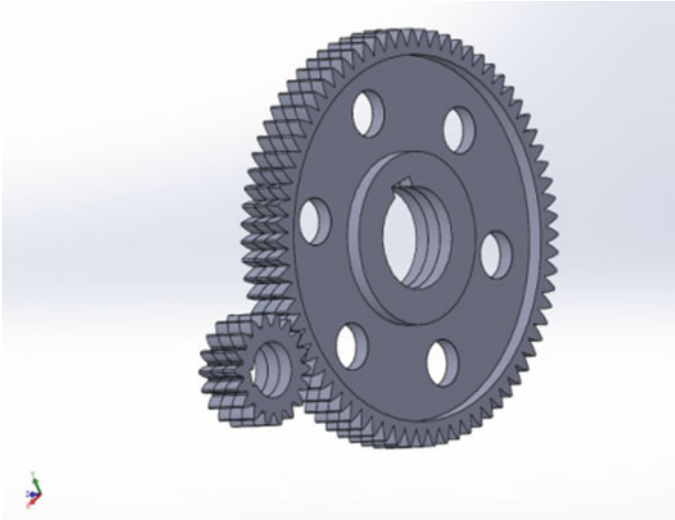
Fig. 3 Sandwich-type CAD model

2.3 CAD Modeling

CAD model has been developed from the optimum dimensions obtained from the PSO algorithms as presented in Table 3. These models have been developed to check the geometric interference of the gear set. Three composite gear sets involving different material and their location in the composite gear set have been developed in this work. These gear sets are sandwiched to one another in three combinations (as can be seen in Table 1). Figure 3 depicts the sandwiched gear set.

2.4 FEM Analysis

The FEM technique plays a vital role in pre-processing, analysis of the geometry configuration of the machinery part. It is the time required to produce a machinery assembly and the examination of the mechanism with its variable aspect. Hence by using the FEM analysis, we can reduce the time and money by plotting the geometry configuration. In SOLIDWORKS, we have modeled the machinery part with its geometry configuration and imported it in ANSYS.

The solution to several engineering problems which have complex models such as the optimum design of spur gear set is not always possible through an analytical approach. For this reason, numerical methods provide approximate but acceptable solutions. The validation of the optimum design of the gear set is performed through stress analysis using FEA. After the validation of the design, the CAD model may transfer to the shop floor for manufacturing.

In pre-processing stage, the geometry and mechanical properties of the structure were defined then discretization of the structure is performed through mesh generation of the whole structure. The mesh's type and element size play a pivotal role in the evaluation of contact stress at the tooth mesh contact.

In this analysis mainly three types of discretization methods (automatic, hex-dominant, and tetrahedral) are used to obtain more reliable solutions as can be seen in Fig. 4. The geometry and mechanics of the model are the two most critical parameters for meshing. Simple regions are meshed using hex-dominant meshing. Tetrahedral meshing is adapted to generate a mesh in complicated regions. An automatic mesh is a hybrid of the two meshing processes. Tetrahedral elements may fit better in complex shapes. Nevertheless, when it integrates the shape functions with points of Gauss it is less precise than hex-dominant elements [15]. Besides, element distortion is an important aspect that governs the mesh quality. This is because of the mapping of integration space from real to natural. To summarize, if the geometry is easy, hexahedral elements are the best choice for meshing. If this is not an option, (acute angles, curved geometries, or similar) then go with tetrahedral but control the distortion of the elements [15]. Hex-dominant meshes are economical with the number of elements due to the constant degrees of freedom for one Hex-dominant refers to six Tetrahedral. Increasing the number of elements will not enhance the size of the global finite element matrices but the computations for one hexahedron are generated also for six tetrahedra. This step has to be compared in computational time to state if it is interesting to use hex-dominant than constant strain tetrahedral knowing that curved or linear hexahedra use Gauss integration points to generate the element characteristics (stiffness, mass) and tetrahedral use exact formula without any integration to get the same characteristics. Tetrahedral components are commonly used by academics because they fit well with arbitrary-shaped geometries and have a low computational cost [15]. The combination of fine and coarse grain size helps the designer to get more precise results. However, the computational cost for fine-grain is more as compared to coarse grain size.

In the solver step, boundary conditions were imposed as the pinion was subjected to a torque (52.7 N-m) along its axis and it is subjected to a fixed support at the end of the shaft. Lastly, in post-processing, analysis results for the maximum stress at the tooth surface were obtained. The solutions of FEA in workbench 15.0 for gear

(a)　　　　　　　　　　(b)　　　　　　　　　　(c)
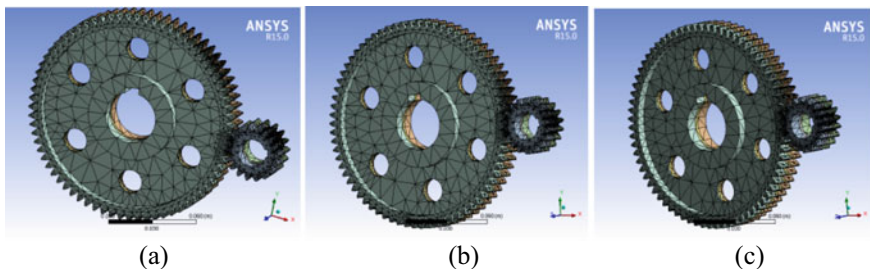
Fig. 4 Gear mesh, **a** Automatic, **b** Hex-dominant, **c** Tetrahedral

sets with different material combinations and meshing types have been reported in upcoming sections. In the optimization, the contact length (T_1T_2) was found to be 46.53 mm. For contact stress analysis, the contact zone that needs to be discretized is usually double the calculated contact length [16]. As a result, the length of contact on both side of the pitch point was determined to be 93.06 mm, and the region was divided into different nodes (14 nodes). As a result, a pitch length of 93.06 mm was divided into 12 parts.

3   Result

In this article, FEM analysis of the sandwiched spur gear and pinion assembly has been performed. The CAD model of the spur gear set has been developed by the optimized dimension obtained through the PSO algorithm. In this study, we have arranged three single-stage spur gear set by varying the face width of the sandwich gear set. Also, the position of the selected gear material is interchanged for the contact stress analysis.

The optimized dimension of the spur gear set obtained through the PSO algorithm is analyzed and CAD models of different sets i.e., set 1, set 2, and set 3 with the considered gear material combinations (combination 1, combination 2, and combination 3) have been prepared in SOLIDWORKS. Finally, these CAD models have been imported into the ANSYS-15 workbench where static structure analysis was carried out.

3.1   Contact Stress Analysis

The optimized gear set is transferred to the FEM workbench for the contact stress analysis. Table 3 presents the values of the maximum contact stress obtained for different gear set arrangements and the material configuration.

It is found that the spur gear set of combination 3 of set 3 has minimum contact stress as compared to other gear sets in all mesh types i.e., automatic, hex-dominant, and tetrahedral mesh types. A comparison of the stress analysis results for the optimized gear set is reported in Table 4 for different gear sets and combination types. The reported results in Table 4 divulges that the automatic mesh type has the minimum contact stress for all the combinations in set 1 and set 2 (except tetrahedral mesh type has minimum stress value in combination 2 of set 2). Whereas, tetrahedral mesh type has minimum contact stress value for all combination types of set 3. On the other hand, Fig. 5 presents the average contact stress variation of the gear set with material combination variation. It can be observed that set 2 has maximum contact stress for combinations 1 and 2. Set 1 has minimum contact stress with combinations 1 and 2. Whereas, for combination 3, set 1 has maximum contact stress and set 3 has minimum contact stress. On the other hand, the designer is free to select the gear set

Table 4 Contact stress (MPa)

Mesh type	Set 1			Set 2			Set 3		
	Combination			Combination			Combination		
	1	2	3	1	2	3	1	2	3
Automatic	237.40	237.28	338.94	520.38	545.16	220.86	345.53	327.44	217.68
Hex-dominant	283.10	282.93	428.79	522.38	540.16	402.85	341.53	327.44	216.68
Tetrahedral	262.92	261.92	427.22	461.56	538.16	390.85	338.53	311.72	215.68
Average	261.14	260.71	398.32	501.44	541.16	338.19	341.86	322.20	261.68
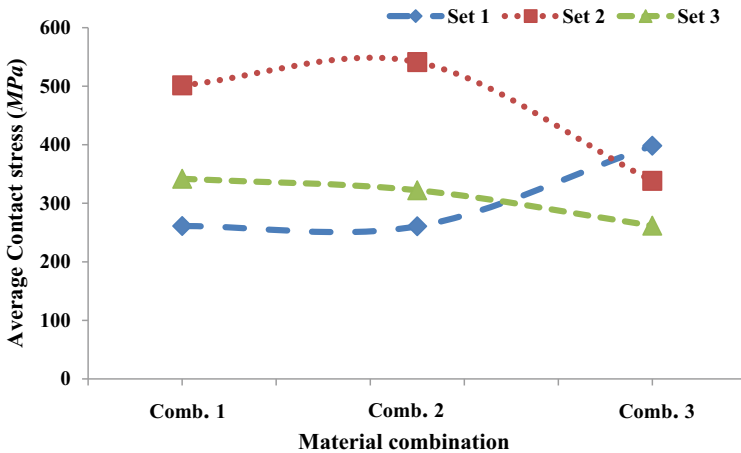
Fig. 5 Average contact stress variation with combination

combination as per their operational requirement and material availability. Figure 6a depicts the contact stress analysis of set 3 combinations 3 with automatic mesh in ANSYS.

3.2 Deformation Analysis

This study presents the deformation of the gear tooth. It is observed that the spur gear set of combination 3 of set 3 has minimum deformation as compared to other gear sets in all mesh types i.e., automatic, hex-dominant, and tetrahedral mesh types. A comparison of the deformation analysis results for the optimized gear sets is reported in Table 5 for different gear sets and combination types. The reported results in Table 5 divulge that combination 1 of set 1 has minimum deformation as compared to another combination type of this set of gear. On the other hand, combination 3 shows minimum deformation under loading conditions for set 2 and set 3. Figure 7 presents the average deformation variation of the gear set with material combination
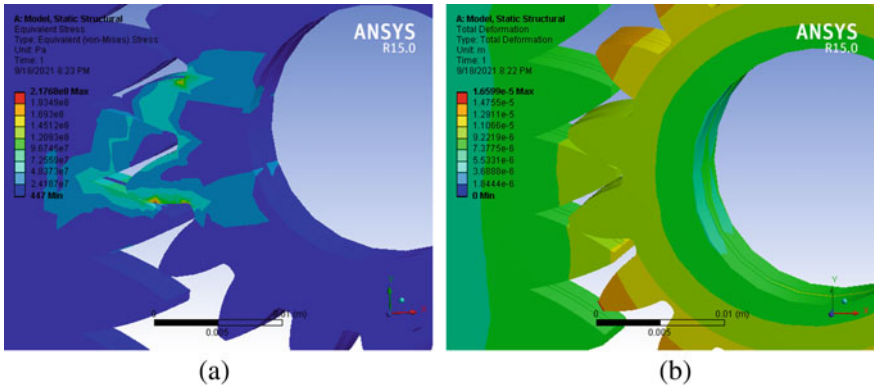
Fig. 6 Set 3 combination 3, **a** contact stress (Automatic mesh), **b** Deformation (tetrahedral mesh)

Table 5 Total deformation ($\times 10^{-5}$ m)

Mesh type	Set 1			Set 2			Set 3		
	Combination			Combination			Combination		
	1	2	3	1	2	3	1	2	3
Automatic	2.38	2.42	5.07	3.00	2.92	2.24	2.98	2.97	1.66
Hex-dominant	2.38	2.43	5.03	3.00	2.92	2.43	2.97	2.97	1.64
Tetrahedral	2.36	2.40	4.94	3.82	2.92	2.43	2.95	3.03	1.66
Average	2.37	2.42	5.01	3.27	2.92	2.37	2.97	2.99	1.65
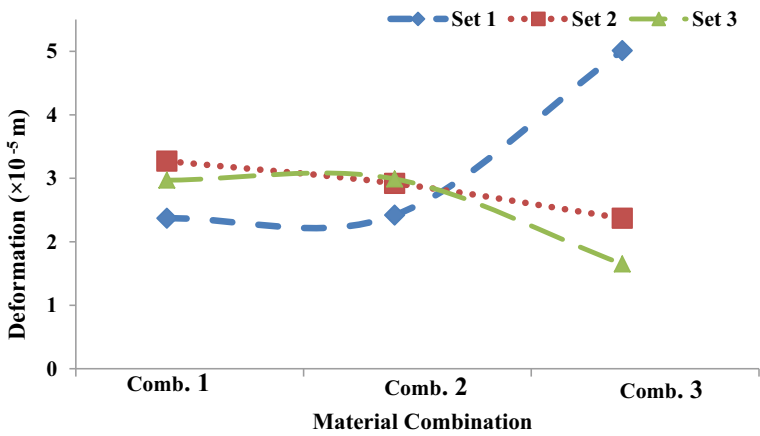
Fig. 7 Average deformation variation with combination

variation. It can be observed that set 2 shows maximum deformation in combination 1 and 2. Set 1 has minimum deformation with combinations 1 and 2. Whereas, for combination 3 set 1 has maximum deformation, and set 3 has minimum contact stress. On the other hand, the reader is free to select the gear set combination as per their operational requirement and material availability. Figure 6b presents the deformation of combinations 3 of set 3 with tetrahedral mesh.

4 Conclusion

The study aims at the evaluation of deformation and estimation of contact stress of the spur gear set. The authors have observed the variation of contact stress of the spur gear and pinion assembly by arranging the different face width thicknesses of the metallic and composite materials. Three different compositions have been explored to estimate the contact stress of the gear set. The solid gear CAD model was developed in SOLIDWORKS. The FEM analysis has been performed on the composite gear to study the stress distribution. This has been demonstrated that an FEA model can be used to reliably simulate the interaction between two bodies by verifying the impact stresses of two gears in contact. The following conclusions are drawn from the study in hand:

  i.   It is interesting to note that the optimized composite gear sets can withstand high load with a lighter weight of the gear set.
  ii.   The comparison between the different sets of composite gears (using Structural Steel, EN24, and Hylum as the gear material) has revealed that, depending upon the requirements, when minimum weight is the prime requirement, the most desired material is Hylum. If the higher load is to be transmitted then, the desired metal is structural steel and EN24.
  iii. The contact stress analysis presented in this study points toward the fact that set 1 has minimum contact stress in composition 1 and 2 and set 3 has minimum contact stress in composition 3.
  iv.   In the case of sets 1 and 2 automatic mesh types, minimum contact stress is reported in compositions 2 and 3 respectively. Whereas in the case of set 3 tetrahedral mesh type has minimum stress in composition 3.
  v.   Deformation analysis revealed that set 1 has minimum contact stress in composition 1 and 2 and set 3 has minimum contact stress in composition 3.
  vi.   In the case of set 1 combination 1, tetrahedral mesh type reports minimum deformation. And in the case of combinations 3 of set 2, the automatic mesh type reports minimum deformation, Whereas, in the case of combinations 3 of set 3 use of hex-dominant mesh type has reported minimum deformation in teeth. 3.

    vii.   Out of all the cases considered in this study combination, 3 of set 3 is the most preferred set combination has resulted in a lighter gear set with low contact stresses.

  viii.   It has been suggested by the authors that Hylum should be in between structural steel and EN24 to avoid gross deformation end.

References

1. Stanojevic VN, Cvejic I (2003) The analysis of contact stress on meshed teeth's flanks along the path of contact for a tooth pair. Mechan Autom Control Robot 3:1055–1066
2. Hassan AR (2009) Contact stress analysis of spur gear teeth pair. World Acad Sci Eng Technol 34:611–616
3. Markovic K, Franulovic M (2011) Contact stresses in gear teeth due to tip relief profile modification. Eng Rev 31(1):19–26
4. Gupta B, Choubey A, Varde GV (2012) Contact stress analysis of spur gear. Int J Eng Res Technol 1(4):1–7
5. Rani TS, Khalandar TD (2013) Spur gear. Int J Comput Eng Res 03(11):7–12
6. Mahebub V, Kevin V (2014) Comparative finite element analysis of metallic and non-metallic spur gear. IOSR J Mechan Civil Eng 11:136–145
7. Sajad HD, Vivek A, Mohammad JK, Arunish M (2015) Investigation of bending stress on a spur gear tooth at design stage by finite element modeling. Int J Mechan Eng Robot 3:13–18
8. Sachidananda HK, Raghunandana K, Gonsalvis J (2015) Design of spur gears using profile modification. Tribol Trans 58(4):736–744. https://doi.org/10.1080/10402004.2015.1010762
9. Al-Qrimli HF, Abdelrhman AM, Hadi HM, Mohammed RK, Sultan HS (2016) Three dimensional spur gear static contact investigations using finite element method. Modern Appl Sci 10:145–150. https://doi.org/10.5539/mas.v10n5p145
10. Alam J, Priyadarshini S, Panda S, Dash P (2020) Optimum design of profile modified spur gear using PSO, intelligent systems, lecture notes in networks and systems. Springer, Berlin, p 185. https://doi.org/10.1007/978-981-33-6081-5_16
11. Alam J, Priyadarshini S, Panda S, Dash P (2021) A study on design optimization of spur gear set. In: Congress on intelligent systems, advances in intelligent systems and computing. Springer, Berlin, p 1334. https://doi.org/10.1007/978-981-33-6981-8_50
12. Alam J, Panda S (2021) A comprehensive study on multi-objective design optimization of spur gear. Mech Based Des Struct Mach. https://doi.org/10.1080/15397734.2021.1996246
13. Panda S, Biswal B, Jena S, Mishra D (2016) An approach to weight optimization of a spur gear. Proc Inst Mechan Eng Part J: J Eng Tribol 231(2):189–202. https://doi.org/10.1177/1350650116650343
14. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of the IEEE international conference on neural networks (ICNN), Perth, Australia, IEEE Service Center, Piscataway, NJ, vol 4, pp 1942–1948
15. ANSYS Meshing User's Guide (2013) ANSYS, Inc., Canonsburg, PA, USA
16. Vijayarangan S, Ganesan N (1994) Static contact stress analysis of a spur gear tooth using the finite element method, including frictional effects. Comput Struct 51(6):765–770. https://doi.org/10.1016/s0045-7949(05)80016-1

Relativistic R3Y Nucleon–Nucleon Potential: Decay Characteristics of ^{124}Ba Isotope Within the Preformed Cluster Decay Approach

Joshua T. Majekodunmi (ID), **Shilpa Rana** (ID), **Nishu Jain** (ID), **K. Anwar** (ID), **N. Abdullah** (ID), **Raj Kumar** (ID), and **M. Bhuyan** (ID)

Abstract The cluster decay properties of ^{124}Ba is analysed using the recently developed R3Y nucleon–nucleon (NN) potential which was derived from the relativistic mean-field (RMF) approach. The interaction barrier characteristics are compared with the well-known phenomenological M3Y NN potential. The densities for the daughters and clusters are also obtained from the RMF formalism, which is folded with the NN potentials to deduce the nucleus–nucleus potential. The decay half-lives are calculated using the preformed cluster decay model (PCM) in which cluster emission undergoes a quantum tunnelling process across a potential barrier. The scaling factor of Blendowske & Walliser is employed for the estimation of the preformation probability P_0 of the emitted clusters. It is noticed that the half-lives predictions of the M3Y potentials are higher than those of the R3Y at neck-length parameter $\Delta R = 0.5$ fm, being correlated with different barrier properties and an extended repulsive core. Furthermore, ^{12}C manifests the least half-life which indicates the shell closure effect, corresponding to ^{112}Sn daughter. The relationship between cluster masses and the decay parameters is also concisely analysed.

Keywords Cluster decay · Preformed cluster decay model · Relativistic mean-field · Half-life · Neck-length.

J. T. Majekodunmi (✉) · K. Anwar · N. Abdullah
Faculty of Applied and Human Sciences, Institute of Engineering Mathematics, Universiti Malaysia Perlis, Arau 02600, Perlis, Malaysia
e-mail: majekjoe1@gmail.com

S. Rana · N. Jain · R. Kumar
School of Physics and Materials Science, Thapar Institute of Engineering and Technology, Patiala, Punjab 147004, India

M. Bhuyan
Department of Physics, Faculty of Science, Center for Theoretical and Computational Physics, University of Malaya, Kuala Lumpur 50603, Malaysia

Institute of Research and Development, Duy Tan University, Da Nang 550000, Vietnam

1   Introduction

The formation of clusters is most characteristic of unstable atomic nuclei. This is commonly referred to as cluster radioactivity. Particularly, barium isotopes within the mass region $A \approx 130$ are marked with smooth transitions in terms of the shell closure and the nuclear deformation [1]. Its prediction was experimentally validated by Rose and Jones [2] and has led to the discovery of clusters ranging from ^{14}C–^{34}S. Beside the experimental attempts, Poenaru et al. predicted the possibility of cluster emissions from proton-rich parent nuclei having $Z = 56$–72 and $N = 56$–72 corresponding to the formation of double-magic ^{100}Sn daughters via the analytical super-asymmetric fission model (ASAFM) [3]. Subsequently, cluster radioactivity in the trans-tin region has been intensely investigated at Dubna [4] and GSI [5, 6] with a specific focus on ^{12}C probable emission. Yet, the challenge remains that the precise half-lives of such nuclei have not been obtained. This necessitates a theoretical investigation and constitute the motivation for the present study.

To study the emission of ^{12}C and few other probable clusters from ^{124}Ba, we have employed the preformed cluster decay model (PCM) of Gupta et al. [7] which stems from the quantum mechanical fragmentation theory (QMFT), assuming that clusters are pre-born within the parent nuclei before their barrier penetration. The interaction barrier is formed from the interplay between the nuclear and the Coulomb potential. The Coulomb potential is easily obtained while the nuclear potential is usually calculated using phenomenological and microscopic approaches [8]. The R3Y nucleon–nucleon (NN) potential was developed from the relativistic mean-field (RMF) theory for the NL3* parameter set, which is used here in the analysis and compared with the well-known phenomenological M3Y NN potential. Further, in the PCM formalism, the neck-formation arises as fragments separate during the decay process. Hence, the neck-length parameter ΔR assimilates the neck-formation effect and modifies the first turning point R_a as shown in Fig. 1. One of us and collaborators have reported in earlier studies [9–11] that $\Delta R = 0.5$ fm is suitable for cluster decay, however, in this study, the neck-length parameter is varied up to 1.0 fm to explore the properties of the recently developed R3Y which manifest different barrier characteristics as compared to the M3Y potential. The ingredients used in the PCM includes the preformation P_0 and the penetration probability P which are calculated from the Blendowske & Walliser scaling factor and the WKB approximation respectively. The Q-values for the systems under study are estimated from the experimental binding energy data [12]. Both fragments (emitted cluster and daughter nuclei) are regarded as spherical and in the ground state.

2   Theoretical Formalism

The nonlinear RMF Lagrangian density used for the description of the interactions between the many-body system of nucleons and mesons is given as [13, 14]

$$\mathcal{L} = \overline{\psi}_i \left\{ i\gamma^\mu \partial_\mu - M^* \right\} \psi_i + \frac{1}{2}\partial^\mu \sigma \partial_\mu \sigma$$
$$- \frac{1}{2}m_\sigma^2 \sigma^2 - \frac{1}{3}g_2\sigma^3 - \frac{1}{4}g_3\sigma^4 - g_s\overline{\psi}_i\psi_i\sigma$$
$$- \frac{1}{4}\Omega^{\mu\nu}\Omega_{\mu\nu} + \frac{1}{2}m_\omega^2\omega^\mu\omega_\mu - g_\omega\overline{\psi}_i\gamma^\mu\psi_i\omega_\mu$$
$$- \frac{1}{4}\mathbf{B}^{\mu\nu}.\mathbf{B}_{\mu\nu} + \frac{1}{2}m_\rho^2\boldsymbol{\rho}^\mu.\boldsymbol{\rho}_\mu - g_\rho\overline{\psi}_i\gamma^\mu\boldsymbol{\tau}\psi_i.\boldsymbol{\rho}^\mu$$
$$- \frac{1}{4}F^{\mu\nu}F_{\mu\nu} - e\overline{\psi}_i\gamma^\mu(\frac{1-\tau_{3i}}{2})\psi_i A_\mu. \tag{1}$$

Here, all parameters retain their usual meanings [13, 14] and reference therein. The effective R3Y (NL3*) NN interaction plus the single nucleon exchange effect is obtained as

$$V_{\text{eff}}^{R3Y}(r) = \frac{g_\omega^2}{4\pi}\frac{e^{-m_\omega r}}{r} + \frac{g_\rho^2}{4\pi}\frac{e^{-m_\rho r}}{r} - \frac{g_\sigma^2}{4\pi}\frac{e^{-m_\sigma r}}{r}$$
$$+ \frac{g_2^2}{4\pi}re^{-2m_\sigma r} + \frac{g_3^2}{4\pi}\frac{e^{-3m_\sigma r}}{r} + J_{00}(E)\delta(s). \tag{2}$$

Similarly, the well-known M3Y effective interaction plus the exchange effect takes the expression [15]

$$V_{\text{eff}}^{M3Y}(r) = 7999\frac{e^{-4r}}{4r} - 2134\frac{e^{-2.5r}}{2.5r} + J_{00}(E)\delta(s), \tag{3}$$

having the ranges in fm and strength in MeV. The nuclear interaction potential $V_n(R)$ is calculated here within the double folding approach [15] by using both R3Y in Eq. (2) and M3Y in Eq. (3) nucleon–nucleon potential. The expression for the nuclear potential is given as,

$$V_n(R) = \int \rho_c(\mathbf{r}_c)\rho_d(\mathbf{r}_d)V_{\text{eff}}(|\mathbf{r}_c - \mathbf{r}_d + \mathbf{R}| \equiv r)d^3r_c d^3r_d. \tag{4}$$

Here ρ_c and ρ_d are the densities of the emitted clusters (c) and the daughter nuclei (d) respectively. This potential is used to deduce the WKB penetration probability with the preformed cluster decay model (PCM). The decay constant λ and half-life $T_{1/2}$ in the PCM takes the form [16]

$$\lambda = v_0 P_0 P, \quad T_{1/2} = \frac{\ln 2}{\lambda}. \tag{5}$$

To obtain the preformation probability (P_0) for the cluster decay, the phenomenological scaling factor proposed by Blendowske and Walliser [17] is employed in the present work and is given by

$$P_0 = (P_0^\alpha)^{(A_c-1)/3}, \tag{6}$$

where $P_0^\alpha(\text{even}) = 6.3 \times 10^{-3}$ is valid for light clusters with $A_c \leq 28$.

3 Calculations and Discussions

As a representative case, Fig. 1 shows the total interaction potential of $^{124}\text{Ba} \rightarrow {}^{112}\text{Sn}$ $+{}^{12}\text{C}$ as a function of the radial distance R. It entails the individual contribution of the nuclear potentials from the phenomenological M3Y (black dashes) and RMF-based R3Y (blue dashes) interactions as well as the Coulomb potentials V_C (red line) at neck-length parameter $\Delta R = 0.5$ fm. The inset gives a magnified view of the respective first turning points of the total potential. From the figure, it is clear that the R3Y (NL3*) displays a lower interaction barrier as compared to the M3Y. In other words, the R3Y (NL3*) is more attractive than the phenomenological M3Y potential. For the sake of brevity, the three-step penetration of the nuclear interaction potential is not shown here (readers are encouraged to check Refs. [11, 18] for exhaustive details).

Figure 2 gives a clear distinction between the half-life predictions of the M3Y and R3Y interactions at different neck-length. The logarithmic half-lives of different isomeric state of neutron-rich ^{124}Ba is displayed in terms of their corresponding clusters. Each cluster is associated with a different magnitude of $\log_{10} T_{1/2}$ and the deepest minima is found at ^{12}C, corresponding to the neighbourhood of a closed shell daughter ^{112}Sn. As such, the effect of shell closure is revealed. It is worth mentioning that, although it has been earlier established in Refs. [9–11] that a fixed neck-length $\Delta R = 0.5$ fm is suitable for cluster radioactivity, here, ΔR is varied since the recently developed R3Y potential is characterised with different barrier properties. This attempt allows probing the dynamics involved in the R3Y calculation. In all cases, the $\log_{10} T_{1/2}$ prediction of the R3Y turns out to be slightly lower than those of M3Y at $\Delta R = 0.5$ fm. A critical inspection of the figure shows that penetration does not occur for clusters with relatively lower masses (^{12}C and ^{16}O) at $\Delta R = 0.5$ fm with the R3Y potential. Consequently, a higher value of the neck-length $\Delta R = 1.0$ fm is used. Nonetheless, a large difference is noticed as ΔR is increased. It is difficult to draw an inference from this analysis since these systems lack experimental measurement. Yet, we opt to suggest that the discrepancy reflects the different barrier characteristics of the R3Y and M3Y potentials and should be carefully resolved by an appropriate choice of the neck-length parameter.

The difference between the potential at the 1st turning point $V(R_a)$ and the available energy for the decay process (Q-value) is referred to as the driving potential $V(R_a) - Q$. The driving potential is presented in Fig. 3 as a function of the participating clusters. The figure manifests the decay characteristics of the interaction potentials employed. While $\Delta R = 0.5$ fm is suitable for M3Y, it is evident that the potential at the 1st turning point $V(R_a)$ will turn out to be lower than the Q-value, and hence, not satisfying the criterion $V(R_a) > Q$ which is necessary for cluster pen-
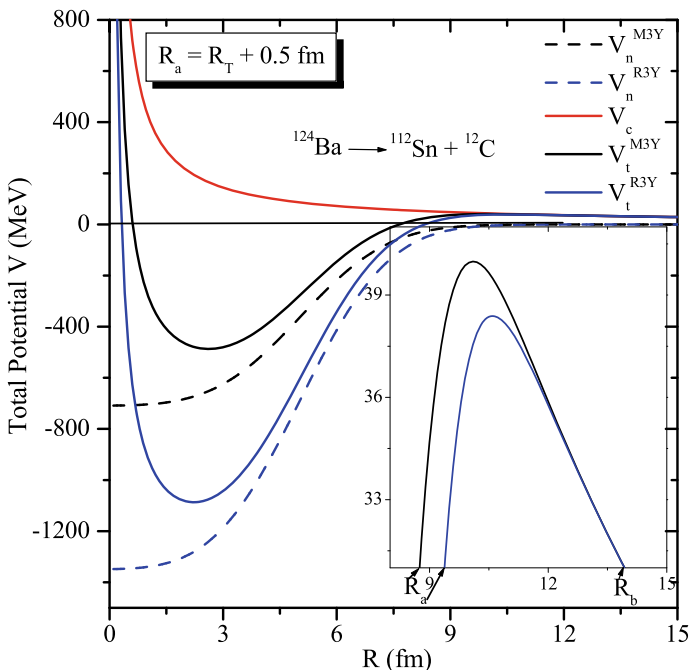
Fig. 1 The profile of total nucleus–nucleus interaction potential V (MeV) and its components, namely, nuclear and Coulomb potential as a function of radial separation for R3Y (NL3*) and M3Y, NN potential for a representative case of ^{124}Ba $\rightarrow$ ^{112}Sn $+^{12}$C. The inset shows a magnified view of the barrier height and position

etration. As such, it is reasonable to increase the ΔR in order to modify its entrance channel and obtain a closer R3Y prediction. More specific details on the precise neck-length which considers the cluster and other participating nuclei within each reaction will be communicated shortly.

Using the Blendowske & Walliser scaling factor in Eq. 6, the cluster preformation factors (given in the 2nd column of Table 1 are found to decrease in magnitude with the respective increase in cluster mass. The Q-values are estimated from the experimental binding energy data [12]. As shown in the 3rd column of the table, the Q-values increase as the cluster mass increases. This implies that there is an inverse relationship between the cluster mass and P_0 while the cluster mass directly imparts the decay energy.

On the other hand, as displayed in columns 4–6, the penetration probability/ penetrability P does not necessarily follow this norm. Instead, it is influenced by the magnitudes of other participating nuclei as well as other inputs within the WKB approximation. From the 5th column, one could notice that the criterion $V(R_a) < Q$ for barrier penetration is not satisfied for the first and second reaction systems at neck-length ΔR = 0.5 fm. This necessitates a modification of the entrance channel by increasing the ΔR to 1.0 fm for the R3Y potential. However, this may lead to certain
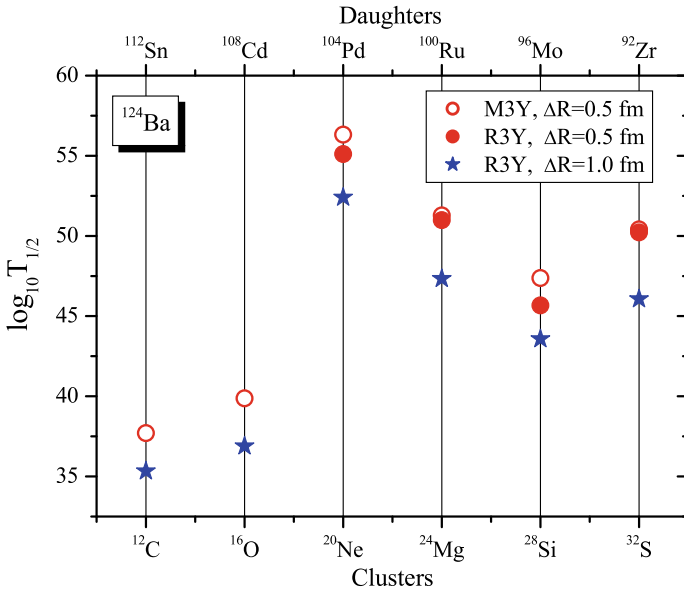
Fig. 2 The logarithmic half-lives at different neck-length parameter ΔR values of ^{124}Ba isotopes for M3Y (open circle) and R3Y (filled symbols)
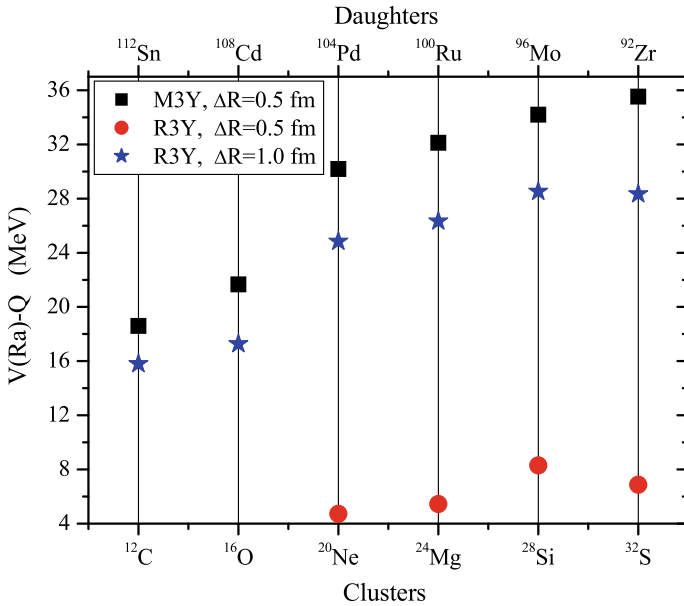
Fig. 3 Variation of the total driving potential $V(R_a) - Q_{RMF}$ at different neck-length parameter ΔR values for ^{114}Ba isotopes. The filled and open symbols are estimated from M3Y and R3Y NN interactions respectively

Table 1 The preformation probability (P_0) from Blendowske & Walliser formula, estimated Q-value obtained from the experimental binding energies [12] and the penetration probability P at ΔR = 0.5 fm. The last column contains the P at ΔR = 0.5 fm for R3Y only.

Cluster mass	Preformation probability P_0 [6]	Q-value (MeV)	Penetrability P		
			M3Y	R3Y	
			ΔR = 0.5 fm	ΔR = 0.5 fm	ΔR = 1.0 fm
12	8.53×10^{-09}	9.5653	7.57×10^{-52}	$V(R_a) < Q$	1.77×10^{-49}
16	9.92×10^{-12}	14.8997	4.07×10^{-51}	$V(R_a) < Q$	3.87×10^{-48}
20	1.15×10^{-14}	17.3473	1.28×10^{-64}	2.10×10^{-63}	1.06×10^{-60}
24	1.34×10^{-17}	24.0712	1.14×10^{-56}	2.24×10^{-56}	9.99×10^{-53}
28	1.56×10^{-20}	31.1979	7.45×10^{-50}	3.78×10^{-48}	4.67×10^{-46}
32	1.82×10^{-23}	35.3848	5.97×10^{-50}	9.12×10^{-50}	1.30×10^{-45}

discrepancies as shown in Figs. (2 and 3). It is therefore reasonable to conclude that the neck-length plays an important role in determining the decay properties within the PCM framework and its suitable value for cluster emissions can be determined in the light of precise experimental measurement.

4 Summary and Conclusions

The half-lives and other decay properties of various isomeric states of the neutron-rich ^{124}Ba isotope have been examined using the relativistic mean-field derived R3Y as well as the phenomenological M3Y NN potentials. We have demonstrated that, unlike the M3Y, a higher neck-length other than the formerly reported value (ΔR = 0.5 fm) could be more suitable for R3Y potential, else, its barrier penetration for clusters with $A_c \leq 16$ could be impeded. The emission of clusters of different masses from ^{124}Ba shows that the inherent properties of clusters (encapsulated in their respective masses) have a strong influence on their individual preformation and decay energy. This study can provide helpful information for future experiments on the probable decay mode of the nuclei in this mass region.

References

1. Gade A, Wiedenhöver I, Meise H, Gelberg A, Von Brentano P (2002) Decay properties of low-lying collective states in ^{132}Ba. Nucl Phys A 697(1–2):75–91
2. Rose HJ, Jones GA (1984) A new kind of natural radioactivity. Nature 245:307
3. Poenaru DN, Greiner W, Gherghescu R (1993) New island of cluster emitters. Phys Rev C 47:2030

4. Oganessian YT, Lazarev YA, Mikheev VL, Muzychka YA, Shirokovsky IV, Tretyakova SP, Utyonkov VK (1994) Search for cluster decay of 114 Ba. Z Phys A 349:341

5. Guglielmetti A, Bonetti R, Poli G, Price PB, Westphal AJ, Janas Z, Keller H, Kirchner R, Klepper O, Piechaczek A, Roeckl E (1995) Identification of the new isotope ^{114}Ba and search for its α and cluster radioactivity. Phys Rev C 52:740

6. Guglielmetti A, Bonetti R, Poli G, Collatz R, Hu Z, Kirchner R, Roeckl E, Gunn N, Price PB, Weaver BA, Westphal A (1997) Nonobservation of ^{12}C cluster decay of ^{114}Ba. Phys Rev C 56:R2912

7. Sharma K, Sawhney G, Sharma MK, Gupta RK (2019) Collective clusterization approach to investigate the relevance of deformation effects in Sn radioactivity. Eur Phys J A 55:30

8. Vorabbi M, Finelli P, Giusti C (2018) Proton-nucleus elastic scattering: Comparison between phenomenological and microscopic optical potentials. Phys Rev C 98:064602

9. Kumar R, Sharma MK (2012) Systematic study of various proximity potentials in ^{208}Pb-daughter cluster radioactivity. Phys Rev C 85:054612

10. Kumar R (2012) Cluster radioactivity using various versions of nuclear proximity potentials. Phys Rev C 86:044612

11. Virk NK, Kumar R, Sharma MK (2020) Cluster radioactivity within the collective fragmentation approach using different mass tables and related deformations. Eur Phys J A 56:35

12. Wang M, Audi G, Kondev FG, Huang WJ, Naimi S, Xu X (2017) The AME2016 atomic mass evaluation. Chin Phys C 41:030003

13. Ring P (1996) Relativistic mean field theory in finite nuclei. Prog Part Nucl Phys 37:193

14. Bhuyan M, Kumar R, Rana S, Jain D, Patra SK, Carlson BV (2020) Effect of density and nucleon-nucleon potential on the fusion cross section within the relativistic mean field formalism. Phys Rev C 101:044603

15. Yahya WA, Falaye BJ (2021) Alpha decay study of Thorium isotopes using double folding model with NN interactions derived from relativistic mean field theory. Nucl Phys A 1015:122311

16. Singh B, Kaur M (2020) Explorations within the preformed cluster decay model. In: Nuclear structure physics. CRC Press, pp 167–183

17. Blendowske R, Walliser H (1988) Systematics of cluster-radioactivity-decay constants as suggested by microscopic calculations. Phys Rev Lett 61:1930

18. Sharma K, Sharma MK (2019) Fragmentation analysis and α-decay half-lives of trans-Sn nuclei. Nucl Phys A 986:1

Predicting Missing Links in Gene Regulatory Networks Using Network Embeddings: A Qualitative Assessment of Selective Embedding Techniques

Binon Teji, Jayanta K. Das, Swarup Roy, and Dinabandhu Bhandari

Abstract Due to the expensive nature and incomplete experimental data, it is always not feasible to experimentally identify exhaustively interrelationship among biological macromolecules. Link prediction is to computationally guess missing relations within a partially constructed network. Representation learning with graph embedding recently achieves great attention towards in-depth network structural analysis and graph entity prediction. In this work, we try to evaluate a few well-known graph embedding techniques for inferring links in gene regulatory networks. We consider *random walk* (RW) and *graph neural network* (GCN)-based embedding methods. It is worth mentioning that RW embedding techniques are not equipped with inherent link prediction capabilities. We try to make the infer missing network links. Experimental results show superior performance by GCN-based methods in comparison with RW-based embedding methods for missing link prediction.

Keywords Link prediction · Graph-structured data · Network embeddings · Representation learning · Graph neural network

B. Teji · S. Roy (✉)
Network Reconstruction and Analysis (NETRA) Lab, Department of Computer Applications, Sikkim University, Gangtok, India
e-mail: sroy01@cus.ac.in

B. Teji
e-mail: bteji.20pdca01@sikkimuniversity.ac.in

J. K. Das
Department of Pediatrics, Johns Hopkins University School of Medicine, Baltimore, MD, USA

D. Bhandari
Department of Computer Science and Engineering, Heritage Institute of Technology, Kolkata, India
e-mail: dinabandhu.bhandari@heritageit.edu

1   Introduction

The biological systems are made up in such a way that the constituent elements, which communicate with each other can be formally structured or modelled as a graph or a network. These networks can be protein–protein interaction, metabolic networks, gene regulatory networks, etc. It is very important to comprehend the activities, associations and relationships that exist within these networks. These objectives can be achieved by modelling these networks mathematically and by employing graph-driven smart data-structural techniques [1].

Gene regulatory network (GRN) is made up of a set of genes that interact with others forming a network. The graph-theoretic formalism of gene networks is the common and simplistic way of representing it.

Definition 1 *(Gene Regulatory Network (GRN))*: A GRN is a graph, $G = (V, E)$, where V is the set of all genes in the network and E is the set of edges between a pair of genes, say $(v_i, v_j) \in V$, representing a strong biomolecular interaction between the two genes.

The edges in a GRN correspond to interdependency between gene activities (nodes). These may include regulation of transcription by transcription factors, but also less intuitive causal effects between genes involving signal transduction or metabolism. Due to the lack of a true network, often they are guessed computationally to recreate such biological networks. The task of reconstructing the network is referred to as *GRN inference*. The development of effective inference techniques is a great challenge in system biology.

High throughput DNA microarray and most recently next-generation sequencing (NGS) technology provide a convenient and effective platform for generating experimental gene expression of a hundred thousand genes simultaneously [1]. The gene expression dataset is represented as an $N \times M$ matrix, $D_{N \times M}$ where each entry $d_{i,j}$ corresponds to the relative abundance of mRNA of a gene, i in jth cell or sample or condition.

A plethora of unsupervised and supervised learning models have been developed in the last few decades [1]. However, they have their limitations in capturing an optimized network. The alternative is to predict missing links from existing partial (inferred or true) networks. Network embedding is a recent development that can be used to predict missing links in a given partially constructed network.

In this paper, we introduce a few network embedding techniques and assess their prediction performance towards inferring missing links in GRN.

2   Network Embedding Techniques for Link Prediction

Link prediction (LP) is the way of predicting the missing link between two nodes in a network or graph-structured data. It turns out to be a very important problem

given any graph-structured data. In gene regulatory network, predicting the missing links in a partial network due to incomplete and noisy data or due to limitations in inference models. Recently, the *graph representation learning* [2] found its niche in learning inherent structural information in a graph. It makes various downstream tasks on graph structure analysis such as node prediction, link prediction, community detection, less challenging, and making them more interpretable in terms of mathematical modelling and computational techniques. It involves the transformation of network nodes or even an entire (sub) graph into lower-dimensional space such that these representations in the embedded space correspond to a notion of similarity between two nodes [3], thereby reflecting the similar inherent structure that exists in the original network. Given, an undirected graph, $G = (V, E)$, with V nodes and E edges, the central aim of graph embedding is to find the mapping $f: V \rightarrow R^d$ from vertices to d-dimensional real vectors.

Literature reports different network embedding methods. We discussed below two popular categories of embedding techniques that will be used next to predict links in GRN.

2.1 Random Walk-Based Embedding

The random walk-based approach for node embeddings has caught much attention because of their expressivity and efficiency. In this method, the first step is to generate a set of node's neighbourhoods for every node via a random walk of the same length. Next, a language embedding model, *SkipGram* [4], which converts graph structure into node embedding in latent space. Below we discuss two random walk-based embedding techniques.

DeepWalk: DeepWalk [5] is a pioneering work that exploits random walk techniques to learn node representations. Due to its capability of employing truncated random walks, it only captures local structural information. Authors define DeepWalk in two steps: primarily, the random walk W_{v_i} is generated for a node v_i as the root node of fixed length (t). The count of generated random walk is denoted by γ. Now, for each node v_i is a generated random walk $|W_{v_i}| = t$. Then, it follows *SkipGram* [4] technique to update the embeddings with the objective function:

$$\underset{\Phi}{\text{minimize}} \ -\log \Pr(\{v_{i-w}, ..., v_{i-1}, v_{i+1}, ..., v_{i+w}\} | \Phi(v_i)) \tag{1}$$

The latter part of embedding updation works by looping over all possible juxtaposition in the random walk that exhibits within the boundary of window w. For a node v_j, a corresponding representation $\Phi(v_j) \in R^d$ is extracted and we maximize the probability of its neighbourhood in the walk. Then, hierarchical softmax [6, 7] approximates the probability distribution by maximizing the probability of a specific path which is represented below:

$$\Pr(u_k | \Phi(v_j)) = \prod_{l=1}^{\lceil log|V| \rceil} Pr(b_l | \Phi(v_j)), \tag{2}$$

where $Pr(b_l | \Phi(v_j))$ is a binary classifier to the parent node b_l, when the path to node u_k is considered as a sequence of tree nodes.

Finally, Stochastic gradient descent (SGD) [8] optimizes the model parameter based on hierarchial softmax.[1]

Node2vec: Node2vec [9] is a more efficient embedding technique that captures not only local but also global structural information of a network. It employs improved *biased random walk* approach also known as 2nd order random walk to generate(sample) node's neighbourhood in BFS *(breadth-first-search)* and DFS *(depth-first-search)* search fashion. Authors define nodes generation with the following strategy:

$$P(c_i = x | c_{i-1} = v) = \begin{cases} \frac{\pi_{vx}}{Z} & if (v, x) \in E \\ \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where Z is the normalizing constant, π_{vx} is unnormalized transition probability between nodes v and x. Here, c_i denotes the ith node in the walk.

In the second-order random walk, there are two variables p and q associated with the walk. The unnormalized transition probability π_{vx} is improved using search bias α with the edge weight $w_{vx} = 1$(for unweighted graphs). More specifically, a random walk that after traversing edge (t, v), currently at node v. Next step in the walk accompanies transition probabilities π_{vx} on (v, x) from v node onwards. The equation setup for unnormalized transition probability goes:

$$\pi_{vx} = \alpha_{pq}(t, x) . w_{wx}, \tag{4}$$

where $\alpha_{pq}(t, x) = \begin{cases} 1/p & if \ d_{tx} = 0 \\ \\ 1 & if \ d_{tx} = 1 \\ \\ 1/q & if \ d_{tx} = 2 \end{cases}$

Here, d_{tx} denotes the shortest path distance between nodes t and x. The parameters p and q are referred to as *return parameter* and *in-out parameter*, respectively, as they supervise how fast the walk explores and leaves the starting node. In the end, the optimization procedure is done using stochastic gradient descent (SGD) [8]

---

[1] Refer to the DeepWalk [5] paper for more information on loss function.

and is done with *negative sampling*[2] which optimizes neighbourhood preservation objective.

The major limitation of random walked-based embedding methods is that its performance largely depends on the topology of the given network. It is less tolerant towards biases, noise and incomplete networks which are common in GRNs. It is important to mention that walk-based methods in their original form unable to predict missing links in a network.

2.2 Neural Network-Based Embedding

Neural network (NN)-based embedding leverages deep learning in graph structures. The implications with the new paradigm have gained a lot of attention in graphs more recently, as it acquires the inductive capability and it can also encode large-scale graphs. Graph convolutional network (GCN)-based learning framework has been exploiting NN embeddings to compute embeddings for the input graph. The best part of GCN is that it leverages the information of the network nodes through parameter sharing using both adjacency matrix(A) and feature matrix(X). Conceptually, a GCN with L layers, $L \geq 2$ with $Z = H^{(L)}$, we get:

$$\begin{cases} H^{(0)} = X \\ \\ H^{(l)} = ReLU(\hat{A}H^{(l-1)}W^{(l-1)}) \ for \ l \in \{1, ..L-1\} \\ \\ H^{(L)} = \hat{A}H^{(L-1)}W^{(L-1)}. \end{cases} \tag{5}$$

$\hat{A}$ depicts symmetric normalized form of A as follows $\hat{A} = D^{-1/2}(A + I)D^{-1/2}$, where D is the diagonal node degree matrix of $(A + I)$ and I is the identity matrix. At each layer, the feature vector of a particular node with its neighbourhood nodes feature vector is aggregated with the nonlinear activation function *ReLU*. Here, $H^{(l)}$ and $W^{(l)}$ denote a node's neighbourhood and weight matrix at l layer respectively. We next discuss two Graph Neural Network-based models for link prediction.

Graph Autoencoders(GAE): GAE [10] is an unsupervised model that is extended from autoencoders [11, 12] to network structures. Basic architecture of graph autoencoders consists of two components: primarily, the encoder part has *graph convolutional network (GCN)* that produces latent vector (compressed representation) z_i of size F, for each node. More specifically, the latent space is a matrix Z of $N \times F$ dimensions, which is the output applied on A and X, i.e. $Z = GCN(A, X)$. Then, the decoder part reconstructs the adjacency matrix $\hat{A}$ from Z. The decoding operation

---

[2] Refer to the node2vec [9] paper for more information on objective function concerning the negative sampling.

is performed through the inner product between latent vectors along with the sigmoid activation function $\sigma(x) = 1/(1 + e^{-x})$. Therefore, the large the inner product between two vectors, say $z_i^T z_j$, the more chances the nodes i and j are linked together. The final reconstructed matrix $\hat{A}$ is achieved as follows, $\hat{A} = \sigma(ZZ^T)$.

GAE is a deterministic model and less flexible. GAE assumes that the distributions of input nodes have all the mass concentrated at one point. Probabilistic models are a lot more precise than deterministic (non-stochastic) models. GAE may not be well-suited for complex and large graphs. We discuss next the probabilistic extension of GAE.

2.3 Variational Graph Autoencoders (VGAE)

A paramount extension in GAE is variational graph autoencoders [10] originated from the idea of VAE [13] to graph structures. This model of probabilistic version has z_i latent variable of length $F \ll N$ for each node, where $i \in V$. These low-dimensional vectors are node embeddings making an embedded space Z for all the nodes. VGAE [10] defines the inference model as follows:

$$q(Z|A, X) = \prod_{i=1}^{N} q(z_i|A, X), \text{ where } q(z_i|A, X) = \mathcal{N}(z_i|\mu_i, \text{diag}(\sigma^2)). \quad (6)$$

The inference equation above is referred to as the encoding part of VGAE. Here, μ is the matrix of mean vectors μ_i, where $\mu = GCN_\mu(A, X)$. Likewise, $log\ \sigma = GCN_\sigma(A, X)$. Then, the generative model works in the decoding part which is defined by the inner product between latent variables as follows:

$$p(A|Z) = \prod_{i=1}^{N} \prod_{j=1}^{N} p(A_{ij}|z_i, z_j), \text{ where } p(A_{ij} = 1|z_i, z_j) = \sigma(z_i^T z_j). \quad (7)$$

Here, A_{ij} denotes the elements of A and $\sigma(.)$ is the sigmoid activation function. Lastly, the variational parameters W_i are optimized as follows:

$$\mathcal{L} = E_{q(Z|A,X)}[log\ p(A|Z)] - KL[q(Z|A, X)\,||\,p(Z)] \quad (8)$$

In the equation above, $p(Z) = \prod_i p(z_i) = \prod_i \mathcal{N}(z_i|0, I))$ is the Gaussian prior used inside Kullback–Leibler [14] divergence between q(.) and p(.) as $KL[q(.)\,||\,p(.)]$. In the end, a full batch gradient descent is performed with *reparameterization trick* [13] during training.

Those GCN-based techniques are more expressive than traditional progression or walk-based methods, and they are expensive due to the iterative model refinement process.

3 Experimentation

We try to evaluate the performance of various embedding methods for predicting missing links in GRNs. As discussed above progression-based methods are not for link, prediction. We use the decoder of GAE based method for predicting links from the latent representations.

3.1 Input Network

We use synthetic GRNs generated using GeneNetWeaver [15] (GNW)[3] related to *E.coli* and *Yeast*. Details about the input network in Table 1.

3.2 Experimental Setup

We try to propose a generic link prediction setup that works with any embedding technique. The proposed framework is illustrated in Fig. 1. In the case of

Table 1 Synopsis of input networks generated using GNW.

Network type	#Nodes	#Edges	#Features
E.coli	100	157	210
E.coli	500	1322	210
E.coli	1000	2019	210
Yeast	2000	5253	210
Yeast	3000	8029	210
Yeast	4000	11323	210
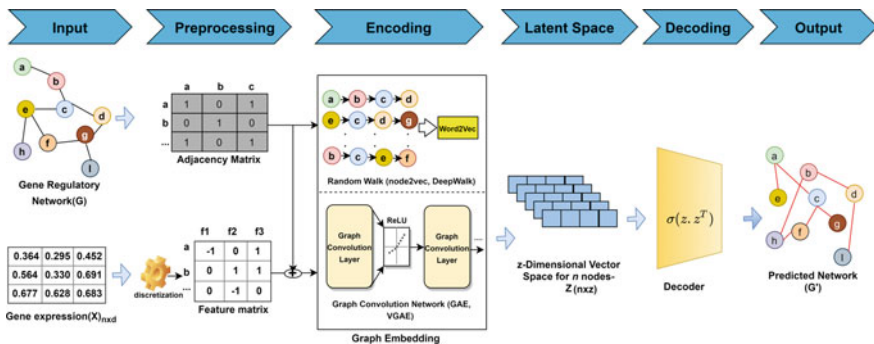
Fig. 1 Architectural pipeline of the experimental setup

---

[3] http://gnw.sourceforge.net/.
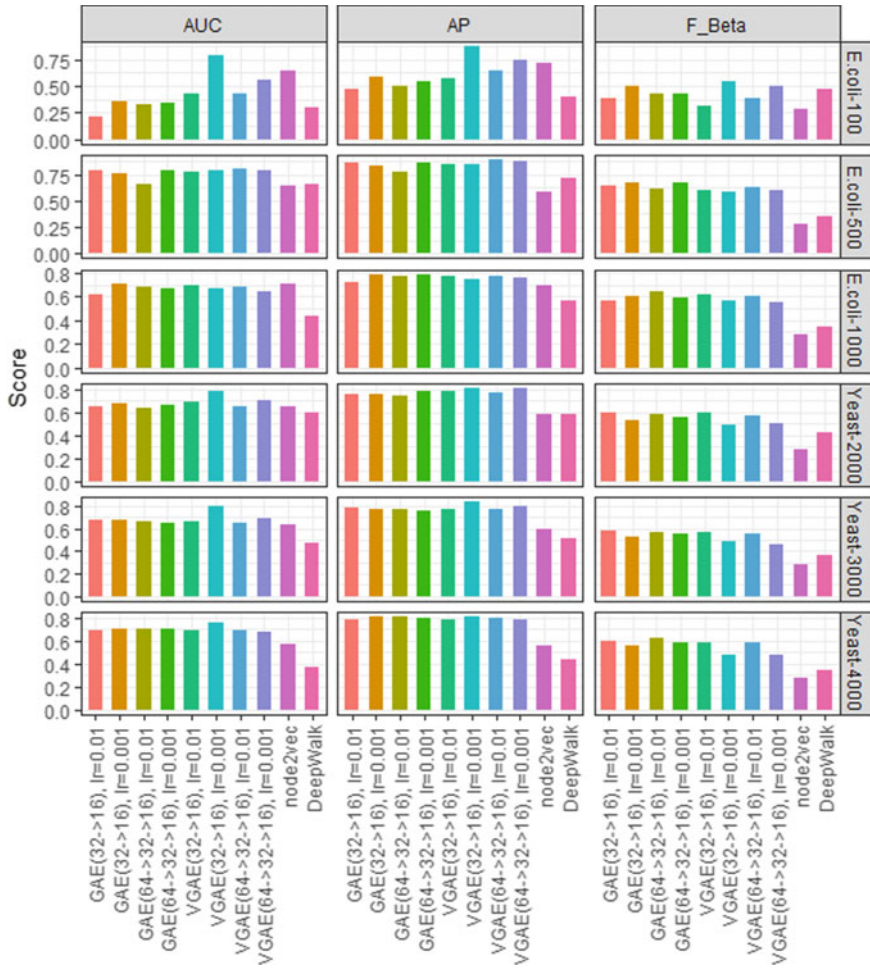
Fig. 2 Trends on embedding methods with different parameters

random walk-based encoding, the input is the edge list of the GRN. GCN-based
encoders accept both GRN and gene expression matrix(nxd), which is commonly
termed as feature matrix. Original gene expression values are first discretized using
`qudiscretize()` method [16] available as R package in *QUBIC*.[4] The decoder
performs the sigmoidal of the dot product between the latent lower-dimensional rep-
resentation matrix, *Z*, with its inverse. The output network represents new links in the
network. Random walk is a single step process, whereas a neural network iteratively
optimizes the weights.

---

[4] https://www.bioconductor.org/packages/release/bioc/vignettes/QUBIC/inst/doc/qubic_vignette.
pdf.

Before encoding, a preliminary step of splitting the edges involves converting an original GRN into a partial version, which is made as input in each of the encoding techniques. The randomly removed edges from the original GRN contribute both to the 5% validation set and 10% test set, respectively. Each of the above sets is assembled by appending missing links from the original network and an equal number of randomly removed edges. For encoding, we have two random walk-based methods: node2vec and DeepWalk, for converting a partial GRN into low-dimensional space and a neural network-based GCN embedding as it appends both partial GRN and pre-processed gene expression discrete matrix for encoding. Hence, GAE and VGAE involve GCN as the default encoder. However, after encoding, the matrix z_i in the latent space is made to be decoded with the dot product $\hat{A}_{ij} = \sigma(z_i^T z_j)$, generating a network with new links.

3.3   Results

We execute node2vec[5] and Deepwalk[6] with default parameter settings (embedding dimensions: 128, context window: 10). For, node2vec and DeepWalk use 80 and 40 length-walk per source, respectively. We record the ability of GAE and VGAE empirically based on the different combinations of hyper-parameter tuning for GCN's hidden layers. We use either 64-dim layers or 32-dim layers and 16-dim layers as latent space, with the learning rate of 0.001 and 0.01 for 200 epochs using *Adam* [17] optimizer. Performance of all the embedding techniques for link prediction is assessed using the metrics *area under the receiver operating characteristic (ROC) curve (AUC), average precision (AP) and F-beta score (beta = 0.5)*. The performance variations with different parameter tuning are reported as a comparison graph in Fig. 2. The overall best performance scores are reported in Table 2.

Experimental results clearly show that neural network-based methods (both GAE and VGAE) achieve superior results in comparison with propagation-based methods towards inferring missing links with varying sized networks. As expected, VGAE is the most effective model that surpasses its performance almost for every network.

---

[5] https://github.com/aditya-grover/node2vec.

[6] https://github.com/phanein/deepwalk.

Table 2 Link prediction performance scores on *E.coli* and *Yeast* networks using various embeddings

Network type	Scores (in %)	Methods			
		Random walk		**Neural Network(GCN)**	
		node2vec	DeepWalk	GAE	VGAE
E.coli-100	AUC	0.64	0.30	0.36	**0.78**
	AP	0.72	0.40	0.59	**0.88**
	F_β	0.28	0.47	0.50	**0.53**
E.coli-500	AUC	0.64	0.65	0.79	**0.80**
	AP	0.58	0.72	0.87	**0.88**
	F_β	0.28	0.35	**0.64**	0.62
E.coli-1000	AUC	0.71	0.44	**0.71**	0.70
	AP	0.70	0.56	**0.79**	0.77
	F_β	0.28	0.35	0.60	**0.62**
Yeast-2000	AUC	0.65	0.60	0.68	**0.79**
	AP	0.58	0.58	0.76	**0.81**
	F_β	0.28	0.43	**0.53**	0.49
Yeast-3000	AUC	0.64	0.48	0.68	**0.80**
	AP	0.60	0.52	0.79	**0.84**
	F_β	0.28	0.37	**0.58**	0.48
Yeast-4000	AUC	0.57	0.37	0.71	**0.75**
	AP	0.56	0.43	0.80	**0.81**
	F_β	0.28	0.35	**0.55**	0.48

Additionally, we also report the training trends in loss and accuracy of GCN encoders for 200 epochs in Fig. 3 and find that loss values for GAE and VGAE have comparable performance as they monotonically decrease on every dataset. However, the accuracy plot of the GAE model for all datasets yields a better trend of training, while the VGAE model fluctuates for all datasets.
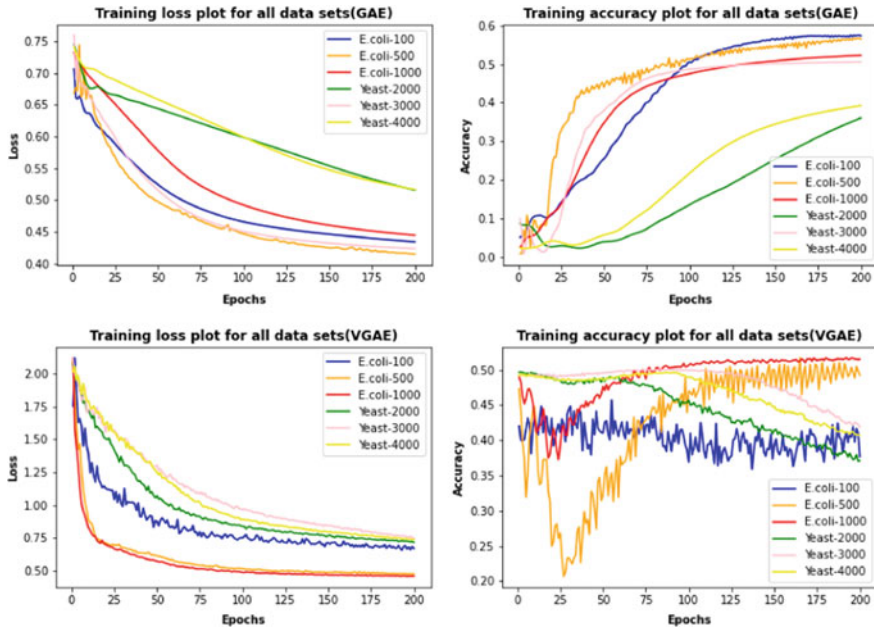
Fig. 3 Train loss and accuracy values for all datasets for GAE and VGAE

4 Conclusion

We present an empirical study on missing link prediction in gene regulatory networks using network embedding techniques, and we propose a framework where any embedding technique (not for link prediction) can be used to infer a link in an incomplete network. Results show that GCN-based methods are superior in performance for predicting links compare to random walk-based methods.

References

1. Guzzi PH, Roy S (2020) Biological network analysis: trends. Graph theory, and algorithms. Elsevier, Approaches
2. Hamilton WL (2020) Graph representation learning. In: Synthesis lectures on artifical intelligence and machine learning, vol 14(3), pp 1–159
3. Mara AC, Lijffijt J, De Bie T (2020) Benchmarking network embedding models for link prediction: are we making progress? In: 2020 IEEE 7th International conference on data science

and advanced analytics (DSAA), pp 138–147. IEEE

4. Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781

5. Perozzi B, Al-Rfou R, Skiena S (2014) Deepwalk: online learning of social representations. In: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, pp 701–710

6. Mnih A, Hinton GE (2008) A scalable hierarchical distributed language model. In: Advances in neural information processing systems, vol 21, pp 1081–1088

7. Morin F, Bengio Y (2005) Hierarchical probabilistic neural network language model. In: International workshop on artificial intelligence and statistics, pp 246–252. PMLR

8. Bottou L et al (1991) Stochastic gradient learning in neural networks. Proc Neuro-Nımes 91(8):12

9. Grover A, Leskovec J (2016) node2vec: scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining, pp 855–864

10. Kipf TN, Welling M (2016) Variational graph auto-encoders. arXiv preprint arXiv:1611.07308

11. Baldi P (2012) Autoencoders, unsupervised learning, and deep architectures. In: Proceedings of ICML workshop on unsupervised and transfer learning. In: JMLR workshop and conference proceedings, pp 37–49

12. Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323(6088):533–536

13. Diederik PK, Welling M (2013) Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114

14. Kullback S, Leibler RA (1951) On information and sufficiency. Ann Math Stat 22(1):79–86

15. Schaffter T, Marbach D, Floreano D (2011) Genenetweaver: in silico benchmark generation and performance profiling of network inference methods. Bioinformatics 27(16):2263–2270

16. Zhang Y, Xie J, Yang J, Fennell A, Zhang C, Ma Q (2017) QUBIC: a bioconductor package for qualitative biclustering analysis of gene co-expression data. Bioinformatics 33(3):450–452

17. Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980

Prediction of Heart Diseases Using Soft Computing Technique

Subhalaxmi Das, Sateesh Kumar Pradhan, Sujogya Mishra, Sipali Pradhan, and P. K. Pattnaik

Abstract In the contemporary lifestyle, the prediction of early health care has a significant character to avoid the death rate caused due to late diagnosis of fatal diseases. To provide better predictive results, many researchers are using machine learning techniques for correct diagnosis. Recently, the data scientist has been paying attention by analyzing big data, which uniquely detects the health status much faster than the expected time and gives a competent approach to conquer the subjects in early detection. Big data analysis is a suitable tool for healthcare prediction using all health parameters to provide better results. Though to classify data is a major hurdle due to its vague nature, the selection of features has a significant role in classifying by eliminating immaterial attributes from the raw data. In our work, we use the Statistical correlation technique to make groups of records with different features; then, using the rough set technique (RST), we find the significant attribute responsible for heart diseases. We validate our claim using statistical techniques.

Keywords Physician data · Rough set theory · Descriptive and predictive data mining

1 Introduction

The heart is the major part of the human physical system. Heart-related diseases are always alarming and not predefined. In the last 20 from 2000 to 2021, significant digital data was applied to various domains such as entrainment, telecommunications, healthcare, and agriculture. A huge number of data has been produced from different fields, various foundations, for example, flooding machines, Instruments

S. Das (✉)
Deparment of Computer Science and Applications, Utkal University, Bhubaneswar, India
e-mail: subhalaxmi.das@gmail.com

S. Mishra · S. Pradhan · P. K. Pattnaik
Odisha University of Technology and Research, Bhubaneswar, India

S. K. Pradhan
Deparment of Computer Science, RBVRR Women's College, Hyderabad, India

with voluminous throughput, networks related to sensors, healthcare by the using various applications of mobile and particularly, the representation of these voluminous data defined as big data [1, 2]. The extraction, envisaging, and abstraction of knowledge by different voluminous data types become difficult, mainly due to the insufficient availability of quality technology. Big data analysis for the identification of important evidence is a significant technological question. Soft Computing methods like Genetic algorithm (GA), Fuzzy set theory (FST), and artificial neural network (ANN) are the fundamental techniques for classifications. The technique of sorting and deriving meaningful information and patterns from voluminous data is defined as data mining, which is useful in deriving meaningful information from previous knowledge [4]. Fluctuation of groundwater level prediction] by using a comparative analysis of artificial neural network (ANN) and optimizing it by the use a hybridization of scale conjugate gradient method, Bayesian neural network and adaptive neuro-fuzzy inference system by Tiwari [5], prediction various diseases by hybridizing rough set theory with recurrent neural network applied on big data analytics by Talasila et al. [6], Kreethika and Premalatha [7] achieved a classification for heart diseases by using a combination of rough set and Fuzzy classifier. Son et al. [8] developed a method to forecast heart failure using a support vector machine, a machine learning tool. Patil et al. [9] discussed the prediction of heart diseases using the support vector machine algorithm; they analyzed the case history data of the patient. Tabesh et al. [10] use a support vector machine to predict heart conditions. They use a support vector machine classifier to increase the accuracy by around 20%. Abraham and Nath [11] work on hybrid intelligent systems design by using artificial neural network (ANN), fuzzy logic (FL) to achieve the optimized result. They used different techniques like a genetic algorithm (GA), simulated annealing (SA). Stephen et al. [12] use a support vector machine for feature extraction with considerable accuracy. Bui et al. [13] discussed on use of time series to predict health care diagnosis on health care issues emphasized on cardiovascular diseases. The work is based on a review of various types of prediction models using time series-based forecasting models. Wijaya et al. [14] had designed a refined detrended fluctuation analysis (R-DFA) to predict heart-bit rate accurately. Rough set is a handy tool for the extraction of significant features from a set of vague and imprecise data, and finding reduct by using the suitable algorithm is a question yet to be answered, Li [15] in his project design a technique by use of decision tree learning algorithm to find an optimal reduct in rough set. Bharti et al. [16] used machine learning and deep learning technique to predict heart disease. They used various machine learning and deep learning algorithm and compared and analyzed their result with the UCI machine learning heart disease dataset. Akulume and Kiwanuka [17] have discussed the correctness of organized performance in forecasting healthcare waste isolation performances. Kaan and İlhana [18] proposed using a genetic algorithm along with a recurrent fuzzy neural network. Li [19] discussed the finding reduct using particle swarm optimization and Johnson's algorithm in a project. Beynon et al. [20] developed a new classification and rule induction method using rough set theory.

The objective of our paper is to look at a number of heart-related disorders that affect people of all ages. The data from numerous medical sources were analyzed using the rough set approach, and it was discovered that heart attacks cause more substantial harm than other heart-related ailments. Our work is focused on the selection of features (symptoms). We utilize a local lower covering method to locate the approximate symptoms of cardiac problems to find the key symptoms.

2 Rough Set Concept and Analysis

The basics of rough set was developed by Poland Mathematician and Computer researcher Pawlak [21] in the early '80 s of twenty century. The basic concept of a rough set is based on basics taken from the information system's logical properties. The rough set concept is a handy tool to apply to various fields like Relational database, Data mining, Knowledge discovery. This theory is loosely based on the concept of Fuzzy set theory. The rough set concept is useful to find structural relationships within imprecise and noisy data.

Fuzzy set and rough set are corresponding ideas of conventional set theory. The rough set concept deals with approximation space with multiple memberships, while the fuzzy set concept deals with partial membership. The growth of both approaches, fuzzy and rough set, strengthens soft computing, the concept of soft computing approximation developed by Zadeh [22], including genetic algorithm, fuzzy set theory, rough set theory, artificial neural network, and all machine learning algorithms. The most common problem on data analysis solved by rough set includes.

2.1 Reduct

The same objects may be signified multiple times. There may be certain attributes that are redundant. We should consider the attributes responsible for the preservation of indiscernibility relation and set approximation. There may be several such sets of attributes called reduct. Construct identical equivalence class set as that articulated by the complete attribute set which can be represented by $[x]_{\text{reduct}} = [x]_p$.

2.1.1 Properties of Reduct

- Reduct sets are minimal.
- There are several reduct sets, so reduct sets are not unique.

$$Core = \cap \text{Re}duct$$

2.1.2   Algorithm to Find the Reduct

Input:

R, set of all conditional attributes, *L*, set of all decision attributes.

Output: *M*, sets with unique feature.

   i. $A := \{\}, \ M := \{\}$
   ii.   repeat
  iii. $A := M$
  iv. $\forall x \in (R--M)$
   v.   if $\gamma_{MU\{X\}}(L) > \gamma_A(L)$
  vi. $A := M \ U \ \{x\}$
 vii. $M := A$
viii.   until $\gamma_A(L) = \gamma_R(L)$
  ix.   return A

There are several algorithms to find the reduct. Finding the reduct is an np-hard problem, i.e., there can be several procedures to find the reduct. So the procedure to find the reduct is not unique.

3   Analyzing Heart-Related Medical Data

Initially, we collected random medical data related to heart diseases which are presented in Table 1. Table 2 represents an information table with respect to heart diseases. The medical data was collected from general public hospitals as well as private hospitals from the districts of Odisha, namely, Khordha, Sambalpur, Cuttack, Jaypore, Koraput, Kandhamal, Baleswar, and Jajpur.

We consider the categories of heart diseases as conditional attributes and, using the concept of minimal attribute concept of rough set theory, find the most significant conditional attribute responsible for the rapid death rate growth. To analyze further, we rename the conditional attributes. Initially, we have more

Table 1 Category of heart diseases

Serial Number	
1	Heart rhythms, or arrhythmias irregularities
2	Marfan syndrome and Aorta disease
3	Heart attack
4	Coronary artery disease (narrowing of the arteries)
5	Deepvein thrombosis and pulmonary embolism

Table 2 Medical data set related to heart diseases

District	Heart rhythms, or arrhythmias ırregularities	Marfan syndrome and aorta disease	Heart attack	Coronary artery disease	Deepvein thrombosis and pulmonary embolism	Total
Khordha	45,000	45,000	15,000	20,000	20,000	145,000
Sambalpur	40,000	65,000	5000	20,000	20,000	150,000
Cuttack	35,000	35,000	2000	21,000	21,000	114,000
Jaypore	35,000	40,000	25,000	16,000	16,000	132,000
Koraput	50,000	25,000	25,000	14,000	14,000	128,000
Kandhamal	25,000	25,000	75,000	14,000	14,000	153,000
Baleswar	25,000	25,000	75,000	16,000	16,000	157,000
Jajpur	75,000	12,000	13,000	5000	5000	110,000

Table 3 Information and correlation

M	R_1	R_2	R_3	R_4	R_5	D
M_1	K_1	K_1	K_2	K_2	K_1	P_1
M_2	K_1	K_1	K_1	K_2	K_1	P_1
M_3	K_2	K_2	K_1	K_1	K_1	P_2
M_4	K_2	K_2	K_2	K_1	K_1	P_2
M_5	K_2	K_2	K_2	K_1	K_2	P_2
M_6	K_1	K_2	K_1	K_1	K_2	P_2

than 200,000 records after using the correlation technique, and we have 6-records, i.e., $< M_1, M_2, M_3, M_4, M_5, M_6 >$ for five conditional attributes $< R_1, R_2, R_3, R_4, R_5, R_6 >$ with their values described as significant and pointless renamed as K_1 and K_2, respectively. The decision attribute single attribute renamed as D, and its values are defined as P_1 and P_2. Table 3 describes the information and correlation between different conditional attributes. Our objective is to find the maximum death rate by which conditional attributes by using rough set theory.

3.1 Finding Reduct and Indiscernibility

Indiscernibility is denoted as IND using an algorithm to find reduct defined in Sect. 3.

$$\text{IND}(R_1) = \{\{M_1, M_2, M_6\}, \{M_3, M_4, M_5\}\},$$
$$\text{IND}(R_2) = \{\{M_1, M_2\}, \{M_3, M_4, M_5, M_6\}\},$$
$$\text{IND}(R_3) = \{\{M_1, M_4, M_5\}, \{M_2, M_3, M_6\}\},$$

$\text{IND}(R_4) = \{\{M_1, M_2\}, \{M_3, M_4, M_5, M_6\}\},$

$\text{IND}(R_5) = \{\{M_1, M_2, M_3, M_4\}, \{M_5, M_6\}\},$

$\text{IND}(R_1, R_2) = \{\{M_1, M_2\}, \{M_3, M_4, M_5\}, M_6\},$

$\text{IND}(R_1, R_3) = \{\{M_1\}, \{M_2, M_6\}, \{M_3\}, \{M_4, M_5\}\},$

$\text{IND}(R_1, R_4) = \{\{M_1, M_2\}, \{M_2, M_6\}, \{M_3\}, \{M_4, M_5\}\},$

$\text{IND}(R_1, R_5) = \{\{M_1, M_2\}, \{M_3, M_4\}, \{M_5\}, \{M_6\}\},$

$\text{IND}(R_2, R_3) = \{\{M_1\}, \{M_2\}, \{M_3, M_6\}, \{M_4, M_5\}\},$

$\text{IND}(R_2, R_3) = \{\{M_1\}, \{M_2\}, \{M_3, M_6\}, \{M_4, M_5\}\},$

$\text{IND}(R_2, R_4) = \{\{M_1, M_2\}, \{M_3, M_4, M_5, M_6\}\},$

$\text{IND}(R_2, R_5) = \{\{M_1, M_2\}, \{M_3, M_4\}, \{M_5, M_6\}\},$

$\text{IND}(R_3, R_4) = \{\{M_1\}, \{M_2\}, \{M_3, M_6\}, \{M_4, M_5\}\},$

$\text{IND}(R_4, R_5) = \{\{M_1, M_2\}, \{M_3, M_4\}, \{M_5, M_6\}\},$

$\text{IND}(R_1, R_2, R_3) = \{\{M_1\}, \{M_2\}, \{M_3\}, \{M_4, M_5\}, \{M_6\}\},$

$\text{IND}(R_1, R_3, R_4) = \{\{M_1\}, \{M_2\}, \{M_3\}, \{M_4, M_5\}, \{M_6\}\},$

$\text{IND}(R_1, R_4, R_5) = \{\{M_1, M_2\}, \{M_3, M_4\}, \{M_5\}, \{M_6\}\},$

$\text{IND}(R_1, R_2, R_4) = \{\{M_1, M_2\}, \{M_3, M_4, M_5\}, \{M_6\}\},$

$\text{IND}(R_1, R_2, R_5) = \{\{M_1, M_2\}, \{M_3\}, \{M_4, M_5\}, \{M_6\}\},$

$\text{IND}(R_1, R_3, R_5) = \{\{M_1\}, \{M_2\}, \{M_3\}, \{M_4, M_5\}, \{M_6\}\},$

$\text{IND}(R_1, R_4, R_5) = \{\{M_1, M_2\}, \{M_3, M_4\}, \{M_5\}, \{M_6\}\},$

$\text{IND}(R_2, R_3, R_4) = \{\{M_1\}\{M_2\}, \{M_3\}, \{M_4, M_5\}, \{M_6\}\},$

$\text{IND}(R_2, R_3, R_5) = \{\{M_1\}\{M_2\}, \{M_3\}, \{M_4\}, \{M_5\}, \{M_6\}\},$

$\text{IND}(R_2, R_4, R_5) = \{\{M_1, M_2\}, \{M_3, M_4\}, \{M_5, M_6\}\},$

$\text{IND}(R_1, R_2, R_3, R_4) = \{\{M_1\}\{M_2\}, \{M_3\}, \{M_4\}, \{M_5\}, \{M_6\}\},$

$\text{IND}(R_1, R_2, R_3, R_5) = \{\{M_1\}\{M_2\}, \{M_3\}, \{M_4\}, \{M_5\}, \{M_6\}\},$

$\text{IND}(R_1, R_3, R_4, R_5) = \{\{M_1\}\{M_2\}, \{M_3\}, \{M_4\}, \{M_5\}, \{M_6\}\},$

$\text{IND}(R_2, R_3, R_4, R_5) = \{\{M_1\}\{M_2\}, \{M_3\}, \{M_4\}, \{M_5\}, \{M_6\}\}.$

The final reduct sets are (R_2, R_3, R_5), (R_1, R_2, R_3, R_4), (R_1, R_2, R_3, R_5) (R_1, R_3, R_4, R_5) and (R_2, R_3, R_4, R_5). The core of the rough set is defined as: $\cap \text{Re}duct$. In this case,

$$\text{Core} = \cap\{(R_2, R_3, R_5), (R_1, R_2, R_3, R_4), (R_1, R_2, R_3, R_5),$$
$$(R_1, R_3, R_4, R_5), (R_2, R_3, R_4, R_5)\} = R_3$$

We concluded that R_3 is the most significant conditional attribute that increases the sudden death rate of R_3 renamed for heart attack. In the subsequent section, we predict the symptoms of heart attack by using different classification techniques.

3.2 Finding Data Analysis Medical Data Related to Heart Attack

In the following Table 4, we have given various symptoms of a heart attack. We intend to find the most significant symptoms responsible for heart attack using the procedure for determining a single local lower covering [23].

We have considered $< 1,2,3 >$ as conditional attributes. Here, R can be a local covering if the following conditions are satisfied:

- $\cup r \in R[r] \subset U$
- For every $r \in R$ one is minimal

    R is minimal $\forall r \in R, \cup_{K \in R-\{r\}}[K][K] \neq \cup_{K \in R}[K]$.

4   Methods to Find Local Lower Covering

This section includes the method to find local lower covering through the following steps:

Input: R ($R \subset X$, X is the universe)

Output: Unitary Local lower covering K of the set R

begin

$M := R$;

$r = \varphi$;

$p = \varphi$;

while $M \neq \varphi$;

begin

$r = \varphi$;

$r_a = \varphi$;

$r_n = \varphi$;

Table 4 Symptoms for heart attack

Serial Number	Symptoms	
1	Discomfort in chest	A
2	The sensation of weakness/light-headed,/faint	B
3	Pain or discomfort in the jaw, neck, or back	C

$r(M) = \{s|[s] \cap M \neq \varphi\}$

while $\{r := \varphi$ or $[r] \not\subset X\}$ and $r(M) \neq \varphi$

begin

select a pair $s = (b_s, b_v) \in r(M)$, such that $|[s] \cap M|$ is maximum

if matching happens, select $s \in r(M)$ with the smallest cardinality $[s]$;

if another matching happens, select the first pair;

$r := r \cup \{s\}$;

$M := [s] \cap M$;

$r(M) := \{s|[s] \cap M \neq \varphi\}$;

if b_s is symbolic $\{$Let V_{bt} is the domain of $a_t\}$

then

$r_s = r_s\{(b_t, v)|v \in V_{at}\}$

else

$\{b_t$ is numerical, $l = (b_t, u, ..., v)$

$r_n \qquad = \qquad r_n \quad \cup \quad \{b_t, x, ..., y|\text{disjoint } x, ..., y \text{ and } u, ..., v\} \qquad \cup$
$\{(b_t, x, ..., y)| x, ..., y \text{ superset } u, ..., v\}$

$r(M) := r(M) - \{r_s \cup r_n\}$;

end (while);

if $[r] \subseteq R$

then

begin

for each b_t (which are numerical attributes) with $(b_t, u, ..., v) \in r$

do

while (r contains at least two different pairs $(b_t, u$ to $v)$ and $(b_t, x$ to $y)$ with equal weight for numerical attributes b_t)

reinstate the two pairs these two pairs with a fresh pair

$(b_t,$ common part of $u, ..., v$ and $x, ..., y)$;

for each t in r do

if $\{r - \{t\}\} \subseteq R$ then $r := r - \{t\}$;

$r := r \cup \{r\}$;

end [then]

Table 5 Information table of heart attack

Records	Discomfort in the chest	A sensation of weakness/light-headed,/faint	Pain or discomfort in the jaw, neck, or back	Decision
A	70–106	Yes	Yes	Yes
B	?	Yes	Yes	Yes
C	65–105	Yes	?	Yes
D	?	No	No	Yes
E	65–95	No	No	No
F	115–118	Yes	No	No
g	95–105	No	Yes	No

else

$p := p \cup \{r\}$;

$M := R - \cup_{K \in r} \cup_p [s]$;

end (while);

every r

do

if $R - \cup_{K \in r} \cup_p [s] = \cup_{K \in r}[s] \Rightarrow r = r - \{r\}$;

end (Procedure);

Using the correlation technique applied on the records given in Table 2, we have an attributes table inclusive of both conditional attributes and decision attributes for heart attack cases. The symptoms for heart attack explained in Table 5 are collected from Table 2.

4.1 Approximation Using Local Lower Covering

Here, universal set $R = \{a, b, c, d, e, f, g\}$ and the conditional attribute defined as $L = \{A, B, C, D\}$, where A: Discomfort in the chest, B: Sensation of weakness/light-headed/faint, C: Pain or discomfort in the jaw, neck, or back, and D: Decision attributes. In general, missing attributes are lost, so it is denoted by (?).

Local lower covering r of R using the method $\cup_{K \in r}[K]$ is a lower approximation for R, Decision explained as [(Heart attack, yes)] = [1, 2, 3, 4] number rows as per the Table 5. In the first instance, we have calculated the block's attribute-value pairs.

$$[\text{Heart} - \text{beat rate, } 75 - 95] = \{e\}$$

[Heart − beat rate, 60 − 107] = {a, c, e, g}

[Heart − beat, 95 − 119] = {f, g}

[Heart − Beat, 65 − 68] = {c}

[Sensation of weakness/light − headed/faint, Yes] = {a, b, c, f}

[Sensation of weakness/light − headed/faint, No] = {d, e, g}

[Pain or discomfort in the jaw, neck, Yes] = {a, b, g}

[Pain or discomfort in the jaw, neck, no] = {d, e, f}

$r(M)$ are relevant attribute-value with $M = \{a, b, c, d\}$ from Table 9 with [Heart-beat rate, 90–107], 105–119, 65–70}], [Sensation of weakness/light-headed/faint, Yes], [Sensation of weakness/light-headed/faint, No], [Pain or discomfort in the jaw, neck,/Yes], [Pain or discomfort in the jaw, neck,/No].

The set [s]∩M is the largest set for s = [Sensation of weakness/light-headed/faint, Yes] = {a, b, c}, choose [Sensation of weakness/light-headed,/faint, Yes] and it is symbolic representation R_a = {[Sensation of weakness/light-headed/faint, Yes], [Sensation of weakness/light-headed,/faint, No} since [Sensation of weakness/light-headed,/faint, No}⊄{a,b,c,f}, entering in the inner while loop $M = \{a, b, c\}$ and $r(M)$ is the set after dropping $r_s ∪ r_n$ from $r(M)$ is [Heart-beat rate, 90–107], 105–119, 65–70}], [Pain or discomfort in the jaw, neck, Yes].

In case of pair values s [Heart-beat rate, 90–107] and [Pain or discomfort in the jaw, neck, Yes].

[s] and M is the largest Also the 2nd attribute the intersection [s] and M is the largest and [s] size are equal for together with its attribute values in pair., we choose the initial one [Heart-beat rate, 90–107] so the class S = [Heart-beat rate, 90–107], [Sensation of weakness/light-headed,/faint, Yes], $[S] = \{a, c\} ⊂ \{a, b, c, d\}$, it is the first nominee for an element for local lower covering is identified in r. Similarly, using the above algorithm iteratively, we generate the following rules.

Our basis $\{a, b, c, d\}$, i.e., Heart attack is yes, The following rules generated by using the above local lower covering is.

Rule-1: K_1-(Sensation of weakness/light-headed/faint, Yes), (Heart-beat rate, 70–106) and (Pain or discomfort in the jaw, neck, or back, Yes) → (Heart attack, Yes).

Rule-2: K_2-(Sensation   of   weakness/light-headed/faint,   Yes) and (Pain or discomfort in the jaw, neck, or back, Yes) → (Heart attack, Yes).

Rule-3: K_3-(Sensation   of   weakness/light-headed/faint,   No) and (Pain or discomfort in the jaw, neck, or back, No) → (Heart attack, No).

Rule-4: K_4-(Sensation of weakness/light-headed/faint, No) and (Heart − beat rate, 65 − 105) → (Heart attack, No).

5 Mathematical Validation

We use the Statistical method to validate our claim as the data collected from different sources are heterogeneous as shown in Tables 6 and 7 and do not follow any particular distribution. Thus we use the Chi-square test to verify our claim.

H_0(Null Hypothesis)-: Heart attack does not depend upon the conditional attributes $< A, B, C>$

H_1 (Alternative Hypothesis)-: Heart attack depends upon the conditional attributes $< A, B, C>$

$$\chi^2 = \sum_i \frac{(E_i - O_i)^2}{E_i} \qquad (4)$$

Table 6 Statistical data observed sample

(Heart attack, symptoms)/Places	A	B	C	D	E	Total
Discomfort in the chest, i.e., (Beat rate)/minute	20	40	20	10	5	95
A sensation of weakness/light-headed,/faint	10	40	40	40	25	155
Pain or discomfort in the jaw, neck, or back	5	10	10	5	5	35
Discomfort in chest/other than Beat rate	5	10	10	10	5	40
Total	40	100	80	65	40	325

with a significance level, $\alpha = 0.05$.

Expected value calculated by the formula $= \frac{\text{Row Total} \times \text{Column Total}}{\text{Grand Total}}$

Table 7 Statistical data expected sample

(Heart attack, symptoms)/places	A	B	C	D	E
Discomfort in the chest, i.e., (Beat rate)/minute	11.7	29.2	23	19	11.7
A sensation of weakness/light-headed/faint	19.07	47.6	38.15	31	19.07
Pain or discomfort in the jaw, neck, or back	4.30	10.7	8.6	7	4.3
Discomfort in chest/other than Beat rate	4.9	12.3	9.846	8	4.9

So calculated value, $\chi_C^2 = 43.76$, where $\alpha = 0.05$ with degrees of freedom, $d_f = $ (Column -1) × (Row -1) $= 3 \times 4 = 12$.

But tabulated value, $\chi_T^2 (1 - \alpha, d_f) = \chi_T^2 (0.95, 12) = 21.03$. Since $\chi_T^2 < \chi_C^2$, we reject the null hypothesis and accept the alternate hypothesis.

6   Comparison with Contemporary Work

Instead of using all of the heterogeneous data for our research, we used statistical correlation to categories the data according to dissimilarity, which provides a better approximation, visualization, and concrete conclusion. b. Using the rough set technique to identify the smallest viable groupings from data that was confusing, imprecise, or absent (information loss). The number of symptoms linked with a heart attack is decreased to an absolute minimum when the Lower Covering Algorithm is utilized. Unlike the cited article [10], which forecast future increases in heart-related anomalies without predicting their symptoms, we predicted the most terrifying heart-related ailments and then discovered their symptoms. Doctors have found our forecast to be quite helpful in the treatment of cardiac illnesses. The majority of the referred work is based on a local heart-related data set, and as the volume of the data rises, the difficulties increase, making generalization impossible. In our research, we focused on getting a general-purpose outcome using ambiguous and imprecise data that was heterogeneous in character, with a focus on a worldwide data set gathered from multiple medical sources.

7   Concluding Remark and Future Work

In this work, we use two concepts of rough set theory, algorithm to find the reduct and the use of local lower covering to generate a set of rules, i.e., a combination of conditional attributes (Symptoms) to generate a set of rules for heart attack. Initially, we had taken several heart-beats-related medical data from the different hospitals, then by correlation method, we had 6-different groups of records, and then we apply the rough set algorithm to find the most significant attribute which increases the death rate. After getting the most significant attribute, we apply a local lower covering algorithm to derive a set of rules for the conditional attribute. This concept can be extended to various fields like Business, Entertainment, and Sports.

References

1. Manogaran G, Lopez D (2018) Health data analytics using scalable logistic regression with stochastic gradient descent. Int J Adv Intel Paradigms 10(1–2):118–132
2. Richter AN, Khoshgoftaar TM (2019) Efficient learning from big data for cancer risk modelling: a case study with melanoma. Comput Biol Med 110:29–39
3. Leong AM, Crighton EJ, Moineddin R, Mamdani M, Upshur REG (2006) Time series analysis of age related cataract hospitalizations and phacoemulsification. BMC Ophthalmol 6(1):2
4. Frawley WJ, Piatetsky-Shapiro G, Matheus CJ (1992) Knowledge discovery in databases: an overview. AI Mag 13(3):57–70

5. Tiwari RK (2013) A comparative study of artificial neural networks, Bayesian neural networks and adaptive neuro-fuzzy inference system in groundwater level prediction. Environ Earth Sci 71(7):3147–3160
6. Talasila V, Madhubabu K, Mahadasyam MC, Atchala NJ, Kande LS (2020) The prediction of diseases using rough set theory with recurrent neural network in big data analytics. Int J Intell Eng Syst 13(5)
7. Kreethika T, Premalatha K (2016) Heart diseases prediction system using optimal rough-fuzzy classifier based on ABC. Asian J Inf Technol 15(3):481–492
8. Son Y-J, Kim H-G, Kim E-H, Choi S, Lee S-K (2010) Application of support vector machine for prediction of medication adherence in heart failure patients health care inform res. Korean Soc Med Inform 16(4):253–259
9. Patil M, Jadhav R, Patil V, Bhawar A, Chillarge G (2019) Prediction and analysis of heart disease using SVM algorithm. Int Res J Eng Technol 6(5)
10. Tabesh P, Lim G, Khator S, Dacso C (2010) A support vector machine approach for predicting heart conditions. In: Johnson A, Miller J (eds) Proceedings of the 2010 ındustrial engineering research conference
11. Abraham A, Nath B (2000) Hybrid ıntelligent systems design—a review of a decade of research. IEEE Trans Syst Man Cybern (Part-C) (2000)
12. Alty SR, Millasseaut SC, Chowienczykt PJ, Jakobssont A (2003) Cardiovascular disease prediction using support vector machines. In: 46th midwest symposium on circuits and systems
13. Bui C, Pham N, Vo A, Tran A, Nguyen A, Le T (2017) Time series forecasting for health-care diagnosis and prognostics with the focus on cardiovascular diseases. In: International conference on the development of biomedical engineering in Vietnam
14. Wijaya R, Rajab TLE, Prihatmanto AS, Mengko RKW (2018) Hypertension heart disease heart rate time series classification using reshape detrended fluctuation analysis. Int J Appl Eng Res 13(1)
15. Li X (2006) Finding optimal reduct for rough sets by using a decision tree learning algorithm. San Jose State University
16. Bharti R, Khamparia A, Shabaz M, Dhiman G, Pande S, Singh P (2021) Prediction of heart disease using a combination of machine learning and deep learning. Comput Intell Neurosci, Article ID 8387680
17. Akulume M, Kiwanuka SN (2016) Health care waste segregation behavior among health workers in Uganda: an application of the theory of planned behavior. Hindawi Publishing Corporation J Environ Publ Health, Article ID 8132306
18. Uyara K, İlhana A (2017) 9th ınternational conference on theory and application of soft computing
19. Li X (2014) Attribute selection method in rough set theory. San Jose State University, Master's Projects
20. Beynon M, Curry B, Morgan P (2002) Classification and rule induction using rough set theory. Willy Online Library
21. Pawlak Z (1982) Rough sets. Int J Comput Inform Sci 11(5):341–356
22. Zadeh LA (1996) Advances in fuzzy systems-applications and theory: Fuzzy sets, fuzzy logic, and fuzzy systems 6:394–432
23. Grzymala-Busse JW, Rzasa W (2010) A local version of the MLEM2 algorithm for rule induction. Fund Inform 100:1–18

An Architectural Framework to Manage Heterogeneous Emergencies

Sarmistha Nanda, Chhabi Rani Panigrahi, and Bibudhendu Pati

Abstract An emergency is a combination of unforeseen circumstances or a state that needs immediate action to get rid of the situation. Most of the emergencies such as cyclone, accident, wildfire, riots, landslides, infectious disease, etc., occur very frequently. In any emergency scenario, affected people get scared and could not be able to take any appropriate action. Timely guidance by the corresponding authorities can minimize the loss of life, environment, and property. In this work, a framework has been proposed that can manage heterogeneous emergencies. Through the proposed framework, when the emergency authorities get information about any type of emergency, they can use the cloud service to interact directly with the affected people. The advantage of this framework is to increase the efficiency of emergency management in terms of scalability, cost, as well as time. It can help the affected people to get rid of the situation by the guidance of emergency-specific experts. A case study is also presented to describe the workflow of the framework.

Keywords Emergency · Cloud computing · Knowledge base · Heterogeneous

1 Introduction

Millions of people are getting affected every year by either man-made or natural disasters [1]. To manage the disasters, various techniques are implemented. It is very sensitive because it has a great impact on human lives, infrastructure, and the economy of a country. So proper and timely management is the major challenge. Automation is the key strategy to manage any emergency situation efficiently in terms of time, cost, and employee turnover. The emerging advancement of the cloud and its increasing

S. Nanda (✉) · C. R. Panigrahi · B. Pati
Department of Computer Science, Rama Devi Women's University, Bhubaneswar, India
e-mail: sarmisthananda@gmail.com

C. R. Panigrahi
e-mail: panigrahichhabi@gmail.com

B. Pati
e-mail: patibibudhendu@gmail.com

availability suggest new approaches where processing is performed remotely [2, 3]. The use of cloud computing helps in increasing the latency and reduces the processing time of the application. 4.66 billion active Internet users are there across the world as of 2021 January, and this is 59.5% of the global population [4]. People are getting affected by multiple emergencies and are not able to get rid of the situation due to a lack of proper management. The crises are putting a negative impact on people and keep them under pressure. During that time, they cannot be decisive, and a group of highly skilled individuals are expected for decision management [5]. It is always required to reduce the economic and casualties loss in an emergency [6].

Keeping this in mind, we thought of an architecture by using which experts can reach the disaster-affected people in a common platform to help them. The people using the Internet-enabled gadget can get the advantage of this and also can help the other nearby affected people. The process starts when an emergency is predicted or occurred without any prior knowledge. It means the mobile users are needed to be tracked based on the emergency location and messages regarding the application to be pushed. One question may arise here, "Is the gadget needs to be much more resourceful in terms of battery life, memory, and processing capability?" The answer is no. The reason is that the emergency-specific application needs to be installed for a while i.e., as and when required, and the computation will be done in the cloud. The information regarding the upcoming emergency or recently occurred emergency can be extracted using a cloud service from the various sources such as social media data, sensor data, image data, etc. [7–9]. Through this web service, the type, location, and severity level of the emergency can be identified. Using these information, the emergencies can be managed with the proposed architecture.

1.1 Motivation

Many applications, such as SURAKSHIT, E^3M, Help Me, etc., are developed to manage emergency situations [10] but these are dealing with only one type of emergency. For any emergency, user has to install that emergency-specific application on their gadgets. It is very difficult for an affected person to search, install, and operate the reliable and authentic application during the emergency. If an automation can be done on top of this to manage multiple emergencies with expert's direct supervision then affected people will not get over-panic.

1.2 Contributions

The contributions toward the work are listed below.

- To propose an architecture for managing heterogeneous emergency.
- A case study is presented to describe the working of the framework.

In this work, the study of related work is done in Sect. 2. In Sect. 3, an architecture to manage heterogeneous emergency is proposed and is described in detail. The description includes the framework architecture, layers available in it, and the algorithms to execute the flow. Finally, Sect. 4 concludes the paper with future scope.

2 Related Work

The emergency occurrence frequency is increasing day by day and proper management is a big challenge. Many applications are developed to address different types of emergencies [11–13]. Authors in [14] address the situation based emergency awareness by using the crisis related social media data. They considered the heterogeneous data where the type of crisis is not known. First, the disaster type was identified using the traditional classifiers, such as Support Vector Machine, Decision Tree, K-Nearest Neighbor etc., with tenfold cross validation technique. Then, an ensemble learning method was implemented with the gradient boosting and Adaboost ensemble learner to detect informative message, relevant message, i.e., witness reports, and tropical classification of messages. They also presented a case study with real world dataset. SURAKSHIT is an smartphone-based application developed by Paul et al. [15]. It uses Geographic Information System (GIS) based data aggregation and to make this system functional; Internet is not required. In this work, authors tried to generate a post-disaster crisis map through off-line crowd sourcing of circumstantial data and distributed using a delay tolerant ad hoc network.

Another Mobile Disaster Management System named as MyDisasterDroid was developed by Fajardo et al. [16]. It determines the optimum routes of different geographical locations using the traveling sales man algorithm. This application was developed using android technology. E^3M is a system for managing the emergency that is proposed by Panigrahi et al. [17]. It focuses on the energy efficiency characteristic. A peer-to-peer connectivity is established by this approach when no suitable network found for communication between mobile devices. Help Me is a self-learning opportunistic ad hoc system proposed by Mokryn et al. [18]. It is a smartphone-based ad hoc communications over WiFi which activates at disaster times. In this system, emphasis is on time and power consumption. It is implemented in Apple's Xcode environment using C/C++/objective-c, Haggle middle ware, SQL Server 2008 R2, and GUI interface builder Web service in NET framework.

3 Proposed Architecture

In this section, the proposed framework named as Architecture to Manage Heterogeneous Emergency (AMHE) is presented.

3.1 Architecture and Notations

The proposed architecture AMHE is shown in Fig. 1, and the notations used are listed in Table 1.

This framework consists of three layers named as Cloud Computation, Knowledge Base, and Application Management. The functionalities of each layer is described as follows.

Cloud Computation: The centralized data center is the main concept behind the cloud computation. The databases and the software can be moved to the cloud and many users can access it [19]. The other advantages of the cloud environment include: (i) It provides resources as per the demand of applications. (ii) It can be accessed from anywhere subject to Internet availability and anything can be exposed as a service. (iii) It also offers the capability to provide 24×7 service. Since there is limited resources in remote location, the centralized cloud can provide software as a service, platform as a service, and infrastructure as a service. In the proposed architecture, and all the required web services are assumed to be available in the cloud and computation is to be done in the cloud.
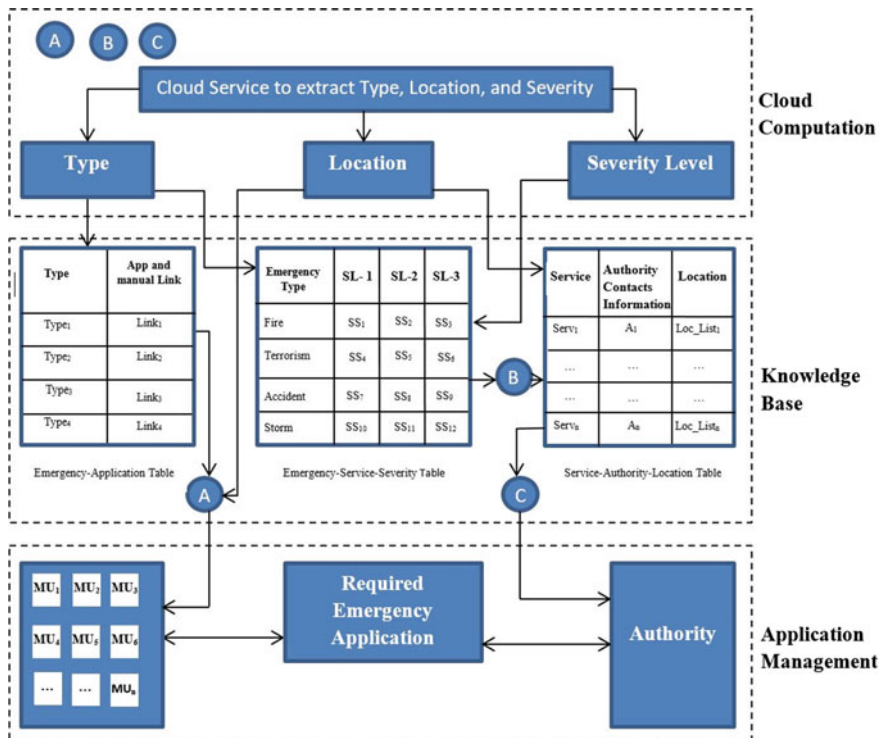
Fig. 1 Proposed architecture to manage heterogeneous emergencies

Table 1 Notations and their definitions

Notations	Descriptions
A	A web service Input: Type, Location Functionality: To search the location-based mobile user, to find the application link, and to send the application link to those mobile users
B	A web service Input: Set of required service Functionality: To identify individual services, to find the corresponding authority and three nearest service availability locations
C	A web service Input: Three nearest services available location, authority information Functionality: To send the location information to the authority
$Type_i$	Emergency Type $\{Type_1, …, Type_i\}$, $1 \leq i \leq 4$
$Link_j$	Mobile Application Link $\{Link_1 … Link_j\}$, $1 \leq j \leq 4$
MU_k	Mobile User $\{MU_1, MU_2, …, MU_k\}$, $1 \leq k \leq n$
SL-p	Severity Level $\{SL\text{-}1, …, SL\text{-}3\}$, $1 \leq p \leq 3$
SS_q	Service Set $\{SS_1, SS_2, …, SS_q\}$, $1 \leq q \leq 12$
$Serv_r$	Individual Service $\{Serv_1, Serv_2, …Serv_r\}$
A_s	Authority $\{A_1, A_2, …, A_s\}$
Loc_List_u	Location List Set $\{Loc\_List_1, Loc\_List_2, … Loc\_List_u\}$

Knowledge base: This is used to store the information and can be accessed as and when required [20]. In this architecture, the knowledge base contains three tables such as *Emergency-Application*, *Emergency-Service-Severity*, and *Service-Authority-Location.* All the required information such as emergency-specific application links with manual in the *Emergency-Application* table, service set required to handle the emergency of different levels in *Emergency-Service-Severity* table, and the service-specific managing authority with service availability location in the *Service-Authority-Location* table are stored in the knowledge base.

Application Management: Through this layer, general public of the affected area will receive the information regarding the emergency-specific application. By installing that application, the affected people and the authority can communicate with each other directly through the Internet-enabled gadgets and proper advice from the experts can reach the people.

The detailed working of the framework is described as follows:

The type of emergency will be used to select the corresponding application and its manual link from the Emergency-Application table available in the knowledge base. Based on the location, the Mobile Users (MUs) will be tracked and a push message will be sent to them regarding the application with its link details using the web service denoted as 'A' in Fig. 1. All the users will be instructed to download that application and can install it on their gadgets and the same link can also be shared and used in any other available, Internet-enabled gadgets. Next, the type and severity

level will be checked in the *Emergency-Service-Severity* table that corresponds to the specific emergency. The set of services will be picked and the individual services such as police force, fire extinguisher, crane, ambulance, doctor facility will be identified. The authority of each service, as well as the three nearest locations, will be found out from the *Service-Authority-Location* table using the web service denoted as 'B' in Fig. 1. That information will be shared with the authority using the web service 'C' in Fig. 1. It is assumed that the authorities will have the application installed in their gadget from which they can communicate with the users in the affected area. The authority and the affected people will receive the secured error-free message as they will communicate directly on a common platform.

3.2 Algorithms

The algorithms used in the framework such as *Message to affected people* (Algorithm 1), *Find required service set* (Algorithm 2), and *Message to authority* (Algorithm 3). The algorithms used in this framework are discussed in detail as follows.

Algorithm-1: *Message to affected people*

1.   Assign: Etype ← Type, Eloc←Location

2.   **for each** $Type_i$ ∈ Type

3.   **if**(Etype= =$Type_i$)

4.   al ← $Link_i$

5.   user[]←**searchUser**(ELoc)

6.   **sendMessage**(al,user[])

7.   break

8.   **searchUser**(Eloc)

9.   user[]← all users ∈ affected area

10.   **return**: user[]

11.   **sendMessage**(al,user[])

12.   **for** all $user_i$ ∈ user[]

              send al to $user_i$

Using Algorithm 1, the affected people can able to know about the application and then install it for proper guidance. In step 1, the emergency type and location are assigned to the variables *Etype* and *Eloc,* respectively. In step 2, *Etype* is searched in the *Emergency-Type* list and once the type is found, the corresponding application link will be stored in the variable *al.* Then *searchUser*() method is called to find the

users available in the affected area and is stored in the user list *user[]*. After tracking them, the application link will be shared by calling the *sendMessage()* method.

Algorithm-2: *Find Required Service Set*

1.   **Assign**: Etype ← Type, Eloc ← Location, Eseverity ←Severity

2.   **For each** $Type_i$ ∈ Type

3.   **if**(Etype=Typei)

4.   **for each** severity

5.   **if**(Eseverity=severityi)

6.   sset ← ss

The Algorithm 3 is used to identify the required services to handle the situation. The type and severity level of the emergency are assigned to *Etype* and *Eseverity* respectivily in the step 1. *Etype* is searched in the *Emergency-Type* list in step 2. In step 3, the *Eseverity* is searched in the *Severity level* list. Then, the required service set denoted as SS is found for the corresponding (*Etype, Eseverity*) pair and is stored in the *sset*.

Algorithm-3: *Message to Authority*

1.   **Assign**: Eloc ← Location

2.   **For each** $serv_j$ belongs to SS

3.   **if**($serv_j$ = = $serv_i$)

4.   Fetch A_i and Loc_List

5.   **sendMessage**(Ai, Eloc, Loc_List[])

6.   **sendMessage**(Ai, Eloc, Loc_List[])

7.   Find 3 nearest location

8.   Send 3 nearest location to Ai

The Algorithm 3 is used to send message to the authority to get the information regarding the services that are to be executed along with their availability locations. In step1, the emergency occurrence location is assigned to *Eloc*. In step 2, for each available service, the authority information A_i and the *service-availability-location* list is identified. In step 3, the *sendMessage()* method is called. In this method, three nearest service availability locations will point out and these information will be sent to the concerned authority for further management.

Case Study: Let us suppose consider four emergencies such as Fire, Terrorism, Accident, and Storm. For each emergency, the application link is stored in the knowledge base and having three levels of severity indicated as Severity Level-1, Severity Level-2, and Severity Level-3. The assumption here is that the risk factor increases

with an increase in severity level. For each pair of emergency and severity level, there is a set of services that are required to be executed. The proposed framework is highly scalable. In justification to this, let the requirement is to add three more emergencies whose applications are already built. For this, we have to add three records in the *Emergency-Application* table specifying the emergency type and the application availability link. Next, the identified required service set will be added against the type and severity level. As we are considering three severity levels for each emergency type, we have to add 3×3 i.e., nine rows in the *Emergency-Service-Severity* table. Then, it is required to extract individual services from the set of services. First, each service will be checked in the *Service-Authority-Location* table. If it is already available as a part of another emergency, then nothing is to be done. Otherwise, the service-specific authority and location availability information need to be collected and respective records are to be added. Once the information is collected, we can add records to the *Service-Authority-Location* table. This process will be repeated for all service types.

4 Conclusions and Future Scope

In this work, authors have proposed an architecture which can manage multiple emergencies. Since the framework is developed using the cloud computing environment and all the computation will be done on the cloud so, the proposed framework will not be a heavy-weight application and it can run on any resource-limited handheld device. A case study is also presented which explains the functionality of the proposed architecture. AMHE will be helpful to the general public as well as to different disaster management authorities to manage heterogeneous emergencies using a single framework. In future, we plan to build a prototype of the proposed system.

In this system, the cloud selection algorithms can be incorporated for providing better communication service. Edge computing concept can be incorporated in the framework to deal with the intermittent connectivity issues of cloud computing. Searching algorithms used in the proposed framework can be optimized to decrease the latency of accessing the services from the cloud.

References

1. Khan A, Gupta S, Gupta SK (2020) Multi-hazard disaster studies: Monitoring, detection, recovery, and management, based on emerging technologies and optimal techniques. Int J disaster Risk Reduction 47:101642
2. Kehoe B, Patil S, Abbeel P, Goldberg K (2015) A survey of research on cloud robotics and automation. IEEE Trans Autom Sci Eng 12(2):398–409
3. Sarkar JL, Panigrahi CR, Pati B, Saha AK, Majumder A (2020) MAAS: a mobile cloud assisted architecture for handling emergency situations. Int J Commun Syst 33(13):e3950

4. https://www.statista.com/statistics/617136/digital-population-worldwide/ Last accessed: 18th Oct 2021
5. Carver L, Turoff M (2007) Human-computer interaction: the human and computer as a team in emergency management information systems. Commun ACM 50(3):33–38
6. Hou LX, Mao LX, Liu HC, Zhang L (2021) Decades on emergency decision-making: a bibliometric analysis and literature review. Complex Intell Syst 1–14
7. Nanda S, Panigrahi CR, Pati B, Mishra A (2021) A novel approach to detect emergency using machine learning. In: Progress in advanced computing and ıntelligent engineering. Springer, Singapore, pp 185–192
8. Huang L, Liu G, Chen T, Yuan H, Shi P, Miao Y (2021) Similarity-based emergency event detection in social media. J Saf Sci Resilience 2(1):11–19
9. Kang B, Choo H (2016) A deep-learning-based emergency alert system. Ict Expr 2(2):67–70
10. Nanda S, Panigrahi CR, Pati B (2020) Emergency management systems using mobile cloud computing: a survey. Int J Commun Syst e4619
11. Phi HL, Hermans LM, Douven WJ, Van Halsema GE, Khan MF (2015) A framework to assess plan implementation maturity with an application to flood management in Vietnam. Water Int 40(7):984–1003
12. Slawik M, Blanchet C, Demchenko Y, Turkmen F, Ilyushkin A, De Laat C, Loomis C (2017) Cyclone: the multi-cloud middleware stack for application deployment and management. In: 2017 IEEE ınternational conference on cloud computing technology and science (CloudCom). IEEE, pp 347–352
13. Miller S, Brewer T, Harris N (2009) Rainfall thresholding and susceptibility assessment of rainfall-induced landslides: application to landslide management in St Thomas, Jamaica. Bull Eng Geol Env 68(4):539–550
14. Pekar V, Binner J, Najafi H, Hale C, Schmidt V (2020) Early detection of heterogeneous disaster events using social media. J Am Soc Inf Sci 71(1):43–54
15. Paul PS, Mehta N, Das AK, Agarwal S, Saha S, NandiS (2019) SURAKSHIT: a smartphone-based application for'localized'GIS data aggregation in absence of internet. In: Proceedings of the 20th ınternational conference on distributed computing and networking, pp 393–396
16. Fajardo JTB, Oppus CM (2010) A mobile disaster management system using the android technology. WSEAS Trans Commun 9(6):343–353
17. Panigrahi CR, Sarkar JL, Pati B, Bakshi S (2016) E 3 m: an energy efficient emergency management system using mobile cloud computing. In: 2016 IEEE ınternational conference on advanced networks and telecommunications systems (ANTS). IEEE, pp 1–6
18. Mokryn O, Karmi D, Elkayam A, Teller T (2012) Help me: Opportunistic smart rescue application and system. In: 2012 The 11th annual mediterranean ad hoc networking workshop (Med-Hoc-Net). IEEE, pp 98–105
19. Wei L, Zhu H, Cao Z, Jia W, Vasilakos AV (2010) SecCloud: bridging secure storage and computation in cloud. In: 2010 IEEE 30th international conference on distributed computing systems workshops. IEEE, pp 52–61
20. Mehla S, Jain S (2021) Development and evaluation of knowledge treasure for emergency situation awareness. Int J Comput Appl 43(5):483–493

Improving Navigational Parameters During Robot Motion Planning Using SOMA Technique

Prasant Ranjan Dhal, Pragyan Kumar Pradhan, Manoj Kumar Muni, Saroj Kumar, and Ansuman Padhi

Abstract Science and technology have progressed in recent years as robots gained their popularity in industrial applications with real-time scenarios. The effective and efficient use of robots in real-time applications become a challenging task for the researchers. Use of intelligent algorithms for trajectory generation with proper motion planning while performing required task is required criterion for robotic agents. The Self-Organizing Migrating algorithm (SOMA) is used in this study to plan optimal paths for many mobile robots in both static and dynamic environments. This technique was simulated in V-REP simulator, and the outcomes have been validated in an experimental platform with real Khepera III robots under laboratory conditions. The simulation and experimental outcomes with very less navigational parameter deviation depict the effectiveness of the implemented intelligent path planning algorithm.

Keywords SOMA · Motion planning · V-REP simulator · Khepera III · Navigational parameters

1 Introduction

Mobile robots gained their applications to perform tedious works in real-time scenarios. To do this path planning is the main area of concern. In this research, Khepra-II mobile robot is considered for both simulation and experimental endorsement. The research focuses on mobile robot navigation in a complicated environment

P. R. Dhal · P. K. Pradhan · M. K. Muni (✉) · A. Padhi
Department of Mechanical Engineering, Indira Gandhi Institute of Technology, Sarang, Odisha 759146, India
e-mail: manoj1986nitr@gmail.com

A. Padhi
e-mail: ansumanpadhi@igitsarang.ac.in

S. Kumar
Robotics Laboratory, Department of Mechanical Engineering, National Institute of Technology, Rourkela, Odisha 769008, India

while avoiding obstacles. For the navigational aspect of mobile robots, a variety of literatures is studied and listed as follows. Rahmanpour and Esfanjani [1] adopted distributed energy-aware communication co-planning strategy for motion planning analysis of networked robotic agents. Kovacs et al. [2] proposed animal motion attribute assisted artificial potential field approach for path planning analysis of MOGI-ETHON mobile robot. Zhong et al. [3] presented novel velocity change space using the concept of hybrid velocity hurdles for trajectory evaluation of mobile robots in unknown dynamic scenarios. Castellini et al. [4] employed POMCP technique with velocity regulation for path planning of mobile robots in complex and industrial like atmosphere.

Song et al. [5] developed and implemented beizer curve hybridized with improved PSO algorithm for hassle free smooth navigation of mobile robots. Malviya and Kala [6] investigated human tracking in 3D environment with lidar sensor and performed social robot trajectory planning. Xing et al. [7] proposed adaptive motion planning strategy for redundant wheeled mobile manipulators. Ahmed et al. [8] presented a highly effective hybridized space deformation assisted trajectory planning of mobile robots in partially known environment. For path planning of dual arm constructed robots, a bidirectional rapidly exploring random tree-based optimization technique based on long short-term memory was used by Ying et al. [9]. For autonomous navigation of mobile robots in uncertain dynamic complicated settings, Chen et al. [10] developed a neural dynamics-based bio-inspired optimization method. Sun et al. [11] carried out mapless trajectory planning using deep reinforcement learning gradient based approach for underwater vehicles. Abkari et al. [12] proposed heuristic based motion and task scheduling approach for bi-manual robots. Zhang et al. [13] addressed a new multiple mode assisted navigational approach for effective planning and tracking scheduling of wheeled mobile robots. In a fixed point rotation environment, an evolutionary algorithm was presented and implemented to study global path planning analysis of a soccer training auxiliary robot [13]. Sun et al. [14] developed an improved RRT and cubic spline approach for smooth and safe navigation of Mecanum-wheeled mobile robot. Various artificial intelligence algorithms for robot path planning in diverse terrain were presented by Muni et al. [15–23] and Kumar et al. [24–29].

Though several researchers developed various intelligent algorithms towards motion planning aspects in mobile robots, still efficient use of single standalone intelligent technique is limited to address trajectory generation in complex dynamic environments. This paper investigates the Self-Organizing Migrating algorithm for successful motion planning of mobile robots in complex situations with dynamic barriers.

2 Framework of SOMA

In 2004, Zelinka [30] proposed the Self-Organizing Migrating Algorithm, which is based on the cooperative competitive behaviour of intelligent organisms attempting

to solve a common problem. When an animal group is looking for food, for example, they will usually cooperate as well as compete. When one of the members succeeds, the other members in the group will shift their paths to follow the successful member. If any other member outperforms the previous best by locating more food, all members' paths will be redirected to the new most successful member. This cycle continues until everyone is gathered around a single food source.

2.1 SOMA Operations

SOMA operates in loops, which are referred to as Migration loops. In SOMA, no new offspring are produced as a result of the parents crossing. so new positions are determined for those approaching the current Leader. Migration refers to how they travel around the landscape-hyper-plane. Migration can also be considered as a process of competition and cooperation. During competitive process, each individual efforts to find best place on their path, as well as the best of all the others. As a result, through migration, each individual competes with each other. When everyone gets a new position, they inform others about their cost value. This is a cooperative process. All individuals work together to choose the leader.

In begining SOMA's parameters, such as Path Length, Step, PRT, Pop Size, Min Div, and Migrations, as well as the fitness function, must be specified. A Specimen, which will be used to generate the population, must be defined.

$$Specimen = \{\{Real,\{Lo,Hi\}\},\{Integer,\{Lo,Hi\}\},......\} \tag{1}$$

where,

Lo   Lower border
Hi   Upper border

Population is generated randomly across the entire search space by the following equation.

$$P_0 = x_j^{(Lo)} + rand_j[0, 1] \times \left(x_j^{(Hi)} - x_j^{(Lo)}\right) \tag{2}$$

where

P_0 Initial population.
$x_j^{(Lo)}$ Lowest value for the boundary.
$x_j^{(Hi)}$ The highest value for the boundary.
$rand_j$ a number between 0 and 1 at random.

Individuals of the population are then evaluated using the given fitness function [31].

$$f_{\text{value}} = (a_{\text{atr}}d_{\text{tar}}) + \left(b_{\text{rep}} \sum_{0}^{n_{\text{obs}}} e^{-(c-s_{\text{obs}})d_{\text{obs}}}\right) \tag{3}$$

where

f_{value}	Fitness value.
a_{atr}	The equilibrium coefficient of attractive force.
d_{tar}	The distance from robot to target.
b_{rep}	The equilibrium coefficient of repulsive force.
n_{obs}	The number of obstacles.
c	The influential coefficient of obstacle.
s_{obs}	The size of obstacle.
d_{obs}	The distance from robot to obstacle.

After being evaluated, the individuals of population are represented in the form of a matrix, with each column containing data about the individual: the first row contains fitness values, and the subsequent rows contain the individual locations. After that, the algorithm goes through migration loops. The leader is the person with the highest fitness value, and be selected for the migration loop. The rest of the individual will then move in a step-by-step towards the Leader until the distance between their current and original positions is greater than the Path Length.

In mutation process, SOMA uses the PRT (perturbation) parameter to achieve perturbation movements. It has a value between 0 and 1 and determines whether or not an individual can approach the leader directly. The PRTVector is typically generated before an individual begins their movement across the search space, and it determines the final movement of an active individual in N-k dimensional subspace. Before individuals movement, a random number for each component is generated and comparing it to PRT. If random number generated is less than PRT, the PRT Vector of that jump is set to 1; otherwise it is set to 0. as in the Eq. (4).

$$\text{if rand}_j < \text{PRT then PRT Vector}_j = 1 \quad \text{else } 0, \quad j = 1, \ldots, N \tag{4}$$

After the perturbation movement of the original individual in the search space, the SOMA crossover process generates a group of new individuals known as offspring. The Eq. (5) defines the position of new individuals or offsprings.

$$x_{i,j}^{\text{ML}+1} = x_{i,j,\text{Start}}^{\text{ML}} + \left(x_{L,j}^{\text{ML}} - x_{i,j,\text{Start}}^{\text{ML}}\right) \times t \times \text{PRT Vector}_j \tag{5}$$

where

$x_{i,j}^{\text{ML}+1}$	Individual's new position in the following migration loop.
$x_{i,j,\text{Start}}^{\text{ML}}$	Individual's current migration loop position.
$x_{L,j}^{\text{ML}}$	Leader's position in the current migration loop.
PRT Vector_j	The factor of perturbation.

t   Jumping step, from 0, by Step, to Path Length.

The best position along their travel path of new Individual is selected and compared to the starting position. If it's better, it'll take the place of the initial; otherwise, it'll be skipped.

This process is repeated for all individuals in the population, and once all individuals have finished the jumping process, a new migration loop is initiated. The movement is relaunched after the best new individual is picked as the new leader. The algorithm will continue to execute until the stop conditions are met.

2.2 SOMA Parameters

SOMA is regulated by a collection of unique parameters. Some parameters are responsible for stopping the search process, while others are accountable for the consistency of the optimization process performance. The parameters are listed below.

Path Length: this defines how far an individual follows the Leader. If it's less than 1, the Leader's place hasn't been overshot, which means there's a chance of premature convergence. In that case, instead of finding the global optimum, SOMA may become stuck in a local optimum.

Step size: the granularity with which the search space is sampled is determined by the step size.

PRT: it is an abbreviation for perturbation. This decides whether an individual can move directly to the Leader. The ideal value is close to 0.1. The SOMA convergence speed increases as the value for PRT is increased.

Dim: The dimensionality is determined by the optimization problem.

Pop Size: It refers to the total number of individual in a population.

Migrations: Maximum number of iterations is represented by this parameter. Migrations refer to the essence of SOMA, in which individual beings travel through the landscape in search of the best solution.

Min Div: The Min Div specifies the maximum allowable difference in between the best individual and worst individuals in the population.

2.3 Flowchart of Proposed SOMA

Figure 1 shows the flowchart of the developed SOMA technique.

Fig. 1 Flowchart of SOMA technique
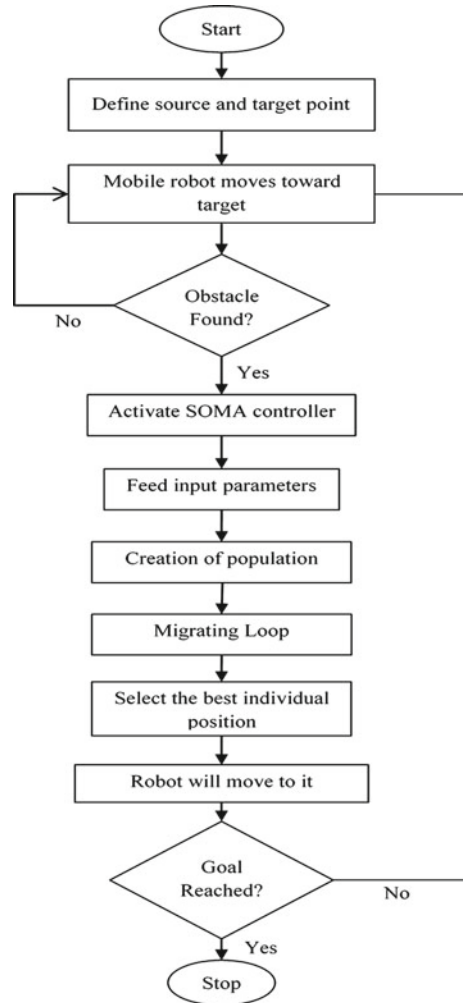
3 Implementation of SOMA for Simulation Analysis

The SOMA technique has been implemented in both simulation and experimental platform on multiple mobile robots to visualize its efficiency. Simulation analysis is done in V-REP software using the SOMA technique in existence of static and dynamic obstacles. The programme is written in LUA language and integrated to V-REP for executing navigation task. In this work, the Khepera III (www.k-team.com) robot has been selected for analysis. The Khepera III is a miniature programmable mobile robot developed by Swiss company K-Team. It measures 130 mm in diameter, 70 mm in height, and weighs 690 grammes. It includes two DC motors along with various infrared proximity and ultrasonic sensors as standard. For running the V-REP

simulation software, we used the Lenovo Ideapad Slim 5i pro machine, which has an Intel 11th Gen Core i5 processor, 16 GB of DDR4 RAM, and 2 GB nvidia MX450 graphics.

The analysis takes into account a 240×160 cm^2 simulation platform with a variety of shaped obstacles. The V-REP platform automatically records navigational parameters such as path length and time.

The designed controller is given to a real Khepera III robot to validate the effectiveness of the SOMA technique. To obtain accurate validation, the same arena size of 240×160 cm^2, size of obstacles, and placement of obstacles are precisely addressed for the experiment setup. The code is written in MATLAB and then executed by mobile robots.

3.1 Navigation Analysis with Multiple Robots

As shown in Fig. 2, the robots are denoted by R1 and R2, as well as their respective starts S1 and S2, and target points T1 and T2. Both robots begin at these points and navigate to their respective targets while avoiding obstacles in V-REP platform. Tables 1 and 2 depict the results from simulation and experimental platforms in relation to navigational parameters.
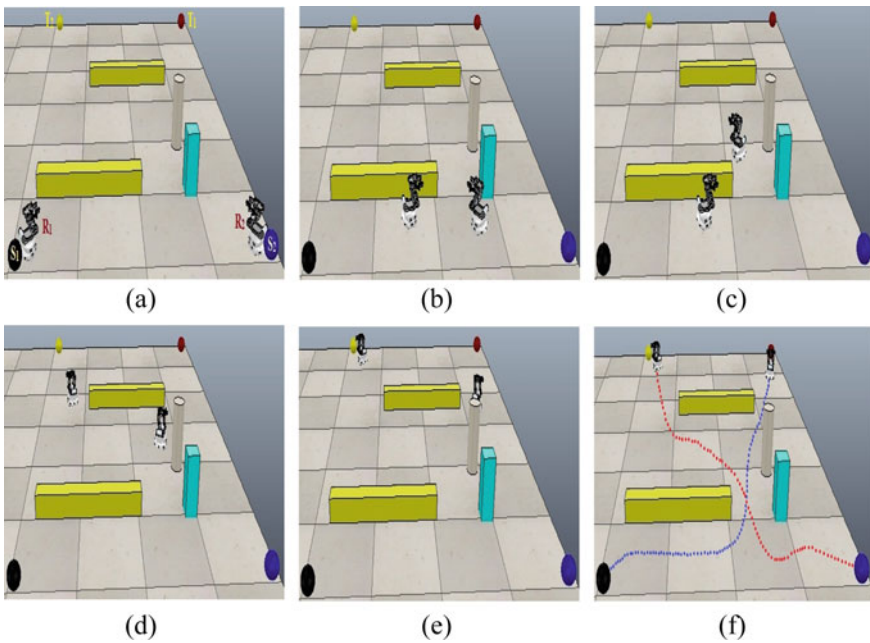
Fig. 2 Navigation of mobile robots in simulated environment

Table 1 Simulation and experimental results of path length for multiple robots

Sl. No	Results simulation		Results experiment		% of deviation	
	Path length (cm)					
	R_1	R_2	R_1	R_2	R_1	R_2
1	314.67	305.43	335.1	325.1	6.09	6.05
2	315.23	305.74	336.2	325.7	6.23	6.12
3	315.16	306.08	336.9	325.9	6.45	6.08
4	314.98	306.19	335.7	326.2	6.17	6.13
5	314.81	305.92	335.8	326.4	6.25	6.27
Average	314.97	305.87	335.9	325.8	6.23	6.13

Table 2 Simulation and experimental results of path time for multiple robots

Sl. No	Results simulation		Results experiment		% of deviation	
	Path time (sec)					
	R_1	R_2	R_1	R_2	R_1	R_2
1	47.12	41.66	50.43	44.52	6.56	6.42
2	47.51	41.72	50.67	44.84	6.23	6.95
3	48.02	42.09	51.28	45.23	6.35	6.94
4	47.83	42.21	51.06	45.24	6.32	6.69
5	48.08	41.88	51.22	45.02	6.13	6.97
Average	47.71	41.91	50.93	44.97	6.31	6.79

4 Conclusions

The current study uses the Self-Organizing Migrating Algorithm to analyze motion planning for numerous robots in dynamic situations. The simulation and experimental analysis are carried out using Khepra-III mobile robot and the outcomes indicate that the proposed algorithm is well enough to perform path planning strategy for multiple robots in complex scenarios as the average percentage of deviation of navigational parameters is found below 7%, which is in acceptable range. The addressed algorithm can be effectively used in real-time and industrial motion planning applications with single standalone technique is the criterion.

References

1. Rahmanpour S, Esfanjani RM (2019) Energy-aware planning of motion and communication strategies for networked mobile robots. Inf Sci 497:149–164
2. Kovács B, Szayer G, Tajti F, Burdelis M, Korondi P (2016) A novel potential field method for path planning of mobile robots by adapting animal motion attributes. Robot Auton Syst

82:24–34

3. Zhong X, Zhong X, Peng X (2014) Velocity-change-space-based dynamic motion planning for mobile robots navigation. Neurocomputing 143:153–163

4. Castellini A, Marchesini E, Farinelli A (2021) Partially observable monte carlo planning with state variable constraints for mobile robot navigation. Eng Appl Artif Intel 104:104382

5. Song B, Wang Z, Zou L (2021) An improved PSO algorithm for smooth path planning of mobile robots using continuous high-degree Bezier curve. Appl Soft Comput 100:106960

6. Malviya A, Kala R (2021) Social robot motion planning using contextual distances observed from 3D human motion tracking. Expert Syst Appl 184:115515

7. Xing H, Torabi A, Ding L, Gao H, Li W, Tavakoli M (2021) Enhancing kinematic accuracy of redundant wheeled mobile manipulators via adaptive motion planning. Mechatronics 79:102639

8. Ahmed A, Maged A, Soliman A, El-Hussieny H, Magdy M (2021) Space deformation based path planning for Mobile Robots. ISA transactions

9. Ying KC, Pourhejazy P, Cheng CY, Cai ZY (2021) Deep learning-based optimization for motion planning of dual-arm assembly robots. Comput Ind Eng 160:107603

10. Chen Y, Liang J, Wang Y, Pan Q, Tan J, Mao J (2020) Autonomous mobile robot path planning in unknown dynamic environments using neural dynamics. Soft Comput 24(18):13979–13995

11. Sun Y, Cheng J, Zhang G, Xu H (2019) Mapless motion planning system for an autonomous underwater vehicle using policy gradient-based deep reinforcement learning. J Intell Rob Syst 96(3–4):591–601

12. Akbari A, Lagriffoul F, Rosell J (2019) Combined heuristic task and motion planning for bi-manual robots. Auton Robot 43(6):1575–1590

13. Zhang X, Huang Y, Wang S, Meng W, Li G, Xie Y (2021) Motion planning and tracking control of a four-wheel independently driven steered mobile robot with multiple maneuvering modes. Front Mech Eng 16(3):504–527

14. Sun Y, Zhang C, Sun P, Liu C (2020) Safe and smooth motion planning for Mecanum-Wheeled robot using improved RRT and cubic spline. Arab J Sci Eng 45(4):3075–3090

15. Muni MK, Parhi DR, Kumar PB, Sahu C, Dhal PR, Kumar S (2021) Global path optimization of humanoid NAO in static environment using prim's algorithm. In: Intelligent systems. Springer, Singapore, pp 25–34

16. Muni MK, Parhi DR, Kumar PB (2020) Implementation of grey wolf optimization controller for multiple humanoid navigation. Comput Animation Virtual Worlds 31(3):e1919

17. Muni MK, Parhi DR, Kumar PB (2021) Improved motion planning of humanoid robots using bacterial foraging optimization. Robotica 39(1):123–136

18. Muni MK, Parhi DR, Kumar PB, Kumar S (2021) Motion control of multiple humanoids using a hybridized prim's algorithm-fuzzy controller. Soft Comput 25(2):1159–1180

19. Muni MK, Parhi DR, Kumar PB, Rath AK (2020) Navigational analysis of multiple humanoids using a hybridized rule base-Sugeno fuzzy controller. Int J Humanoid Rob 17(04):2050017

20. Muni MK, Kumar PB, Parhi DR, Rath AK, Das HC, Chhotray A, Pandey KK, Salony K (2020) Path planning of a humanoid robot using rule-based technique. In: Advances in mechanical engineering. Springer, Singapore, pp 1547–1554

21. Muni MK, Parhi DR, Kumar P, Pandey KK, Kumar S, Chhotray A (2019) Sugeno fuzzy logic analysis: Navigation of multiple humanoids in complex environments. In: International conference on artificial intelligence in manufacturing and renewable energy (ICAIMRE)

22. Muni MK, Kumar S, Parhi DR, Pandey KK (2021) Water cycle algorithm: an approach for improvement of navigational strategy of multiple humanoid robots. Robotica 1–19

23. Muni MK, Parhi DR, Kumar PB, Dhal PR, Kumar S, Sahu C, Kashyap AK (2021) Probability plot result comparison with recurrent neural network approach for path navigation of a humanoid in complex terrain. In: Current advances in mechanical engineering. Springer, Singapore, pp 579–588

24. Kumar S, Parhi DR, Kashyap AK, Muni MK, Dhal PR (2021) Navigational control and path optimization of mobile robot using updated sine–cosine algorithm in obscure environment. In: Current advances in mechanical engineering. Springer, Singapore, pp 989–996

25. Kumar S, Parhi DR, Pandey KK, Muni MK (2021) Hybrid IWD-GA: an approach for path optimization and control of multiple mobile robot in obscure static and dynamic environments. Robotica 1–28

26. Kumar S, Parhi DR, Kashyap AK, Muni MK (2021) Static and dynamic path optimization of multiple mobile robot using hybridized fuzzy logic-whale optimization algorithm. In: Proceedings of the institution of mechanical engineers. Part C: J Mech Eng Sci 0954406220982641

27. Kumar S, Parhi DR, Muni MK, Pandey KK (2020) Optimal path search and control of mobile robot using hybridized sine-cosine algorithm and ant colony optimization technique. Ind Rob Int J Rob Res Appl

28. Kumar S, Pandey KK, Muni MK, Parhi DR (2020) Path planning of the mobile robot using Fuzzified advanced ant colony optimization. In: Innovative product design and intelligent manufacturing systems. Springer, Singapore, pp 1043–1052

29. Kumar S, Muni MK, Pandey KK, Chhotray A, Parhi DR (2019) Path planning and control of mobile robots using modified tabu search algorithm in complex environment. In: International conference on artificial intelligence in manufacturing and renewable energy (ICAIMRE)

30. Zelinka I (2004) SOMA—self-organizing migrating algorithm. In New optimization techniques in engineering. Springer, Berlin, pp 167–217

31. Bao DQ, Zelinka I (2019) Obstacle avoidance for swarm robot based on self-organizing migrating algorithm. Procedia Comput Sci 150:425–432

Review on Automated Detection of COVID-19 from X-Ray Images Using Machine Learning

Debanshu Biswas and Abhaya Kumar Sahoo

Abstract COVID-19 virus has been a worldwide pandemic since its outbreak from December 2019. While coronavirus has a low fatality rate, it is extremely infectious and escalates quickly; therefore, early detection is very important for preventing its outbreak. The procedures currently used by medical personals for detection is RT-PCR test. However, it includes false negative reports and also is a time taking process; thus an alternate solution is required. Any diagnostic system that can detect COVID-19 infection can be very helpful to medical personals. The features found in COVID-19 images by X-rays are very similar to other lung diseases, which makes it very difficult to differentiate. This review includes the contribution of image processing and machine learning to make swift and precise diagnostic system from lung X-ray images. Such a system can be used by radiologists for making decisions and can be very helpful in prior detection of the virus.

Keywords COVID-19 · Image processing · X-Ray · Machine learning

1 Introduction

Coronavirus (COVID-19) is an extremely infectious disease caused by the SARS-CoV-2 virus. Currently, 262,866,050 confirmed cases of COVID-19 were detected according to WHO as of 30 November 2021 [1]. Most individuals affected by Coronavirus are experiencing low to medium respiratory problems and can recuperate without needing any specific therapy, but elderly and people with previous medical issues such as chronic respiratory problems and diabetes. are prone to develop a higher degree of infection due to coronavirus. This virus can also affect the respiratory and cardiovascular systems which may cause many organs to fail or may hamper

D. Biswas (✉) · A. K. Sahoo
School of Computer Engineering, Kalinga Institute of Industrial Technology Deemed to be University, Bhubaneswar, India
e-mail: debanshub7@gmail.com

A. K. Sahoo
e-mail: abhaya.sahoofcs@kiit.ac.in

respiration in same cases. Moreover, the symptoms shown by this virus overlap the usual symptoms which may occur in day-to-day life, which makes it quite challenging to differentiate between COVID-19 from other illnesses. Thus, coronavirus is a global healthcare challenge all over the world.

Most extensively used process for diagnosing coronavirus is the RT-PCR or real-time reverse transcription-polymerase chain reaction test; however it is not sensitive enough and can also give false negative reports which increases the difficulty in detection of the virus. Moreover, it is a time-consuming method and provides comparatively poor detection rate in prior stages.

In 2021, many medical personnel take the following steps to identify COVID-19:

1. Symptoms of the patient are examined such as body temperature, coughing, and breathing problems.
2. RT-PCR test is done.
3. Then, medical personnel determines whether patient has COVID-19 or not. [2].

Some other important diagnostic procedures include:

1. Nucleic acid detection: The requirements of nucleic acid detection is very high and not feasible. Here, quality of the sample can also affect the results. Moreover, this process is very time-consuming.
2. Serum antibody detection: Accuracy is lower than the nucleic acid detection method. For a patient who has suffered COVID-19 previously and has now returned healthy, this result will still give the result as COVID positive [3].
3. Computed tomography (CT): It is also a good alternative of all the methods mentioned above. But the cost is significantly higher than other methods, and the radiation dose of CT scan is much greater limiting the number of times one can go through this procedure.
4. Chest X-Ray (CXR) images: It can obtain quality images with a low cost as compared to CT scan.

Due to all these reasons, previously mentioned methods cannot be used as a standard test for the diagnosis of the virus. Both radiological imaging techniques consisting CT and X-ray are important tool; however, due to the requirement of a radiologist to analyze them, which is a time-consuming process itself and is a setback. Moreover, it is hard to identify the primary features of an illness from CXR images as both COVID-19 and pneumonia produces similar radiological patterns to each other. Therefore, a dataset containing CXR images is most suitable for developing a computer-aided automation system to diagnose COVID-19. Using machine learning techniques, we can improve the diagnosis accuracy of the medical personals. However, radiological scans cannot be a standalone COVID-19 test itself but a supplementary method for confirmation. One of the most common observations in the X-Ray images are the Ground Glass Opacities (GGO), bilateral abnormalities, treads, interstitial abnormalities or stripes, which together with lab tests can be very useful for identifying COVID-19.

Influenced by the studies of machine learning (ML) to develop an artificial intelligence (AI) diagnostic system which can be more accessible and feasible enough to

be widely used, this review includes some of the major contributions made by many researchers for developing a computer-aided system using various machine learning techniques.

2 Literature Survey

With the drastic development of deep learning technology, many researchers have lean toward deep learning, primarily convolutional neural network (CNN) to automate diagnostic healthcare systems. After COVID-19 began infecting people in a global scale, many researchers began studying and analyzing this virus in their respective fields. The accomplishments achieved by deep learning methods to automate the identification of any illness in the healthcare section is major and made it useful for many researchers to design and implement AI systems. Most of them included the identification of COVID-19 using computer-aided diagnostic tools. Many of these researches have been supported by image processing on different radiology scans using deep learning methods. Table 1 shows some automated COVID-19 identification networks by various authors.

3 Methodology

Currently, a large number of computer-aided tools are being developed which uses deep learning architectures to identify many illnesses, which have attracted many researchers and scholars for developing a system for COVID-19. This type of system requires a huge database to produce highest accuracy in its results. Therefore, a well-organized and labeled dataset is required. A schematic illustration of a proposed model is given in Fig. 1.

3.1 X-ray Database

In this step, images of pneumonia, COVID-19 positive and illness-free patients are assembled. The images consist of chest X-Ray scans of all classes mentioned previously, thus creating a huge dataset for training and testing the data. Many researchers have taken the database from open-source platforms such as Kaggle or github, whereas some have created the dataset themselves, such as COVIDx dataset by Wang et al. [4].

Table 1 Work done on automated COVID-19 identification using CXR images

Author	Total images	Network used	Contributions
Wang et al. [4]	13,975	Deep CNN	Designed a special framework called COVID- Net and a dataset called COVIDX
Hall et al. [5]	455	VGG16, ResNet50	Designed an architecture with tenfold cross-validation
Farooq et al. [6]	5941	ResNet50	Created a pre trained ResNet-50 model for identification
Hemdan et al. [7]	50	VGG19, DenseNet201, ResNetV2, InceptionV3	Designed a deep learning architecture called COVIDX-Net
Minaee et al. [8]	5000	ResNet-50, ResNet-18 DenseNet-121, SqueezeNet	Presented sensitivity, specificity, AOC, ROC and confusion matrix for four CNNs
Brunese et al. [9]	6523	VGG16 and transfer learning	Proposed a method to detect region of interest in the CXR which are suggesting COVID-19 infection
Horry et al. [10]	60,798	VGG, Inception, Xception, ResNet	Utilized preprocessed dataset to execute relative testing for many deep learning architectures
Ardakani et al. [11]	1020	SqueezeNet, VGG16, GoogleNet, ResNet50, ResNet101, Xception, MobileNetV2, VGG19, ResNet18, AlexNet	Ten CNNs were utilized to distinguish COVID-19 and non-COVID-19 case
Alhudhaif [12]	1218	ResNet, DenseNet and SqueezeNet	Generalized CNN design was developed to avert any biasing problem
Ahmet Saygili [13]	4353	CNN-SVM, CNN-Sigmoid	Sobel filter is applied to get the contour of images and then fed to CNN and then by SVM classifier
Ghoshal et al. [14]	5941	Drop-weights based Bayesian CNN	Model has been trained by transfer learning on CXR to calculate model uncertainty

3.2 Preprocessing and Image Augmentation

Preprocessing step performs a gray-level conversion for all images and a process of converting the images in an identical format, and all images are then resized to a pre-determined value. In this phase, image sharpening, application of filters such as

Fig. 1 A schematic illustration of a proposed model
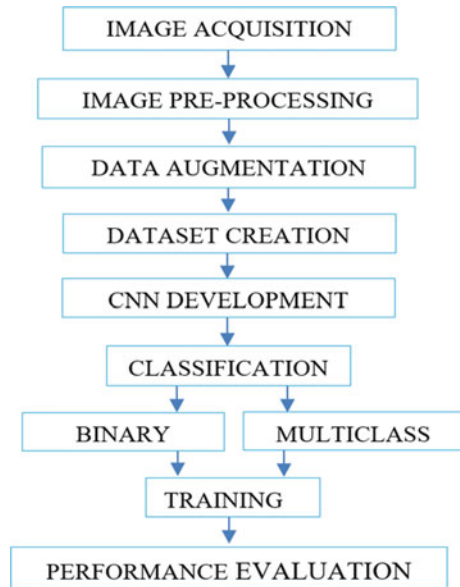
median filter and adjustment of the lighting condition are executed to maximize the precision and accuracy of the image. This process does not reduce the accuracy of the result and reduces transaction cost and time cost [13]. Image augmentation assists in creating more samples by the application of different transformation techniques to the already available dataset images to create some more data for the model training process. Some techniques are horizontal and vertical flips, rotation, shifting, noising, and blurring of the existing image.

3.3 Architecture of Neural Networks

Nowadays, neural networks are vastly employed in many medical applications, with great potential in a various application like object detection, image classification, and medical imaging, including detection of abnormalities in the lungs by CXR images. Convolutional neural networks are one of the most extensively applied deep learning algorithms. CNN do not require any guidance for feature extraction as they are trained from the dataset given. Their design consists of two components: learning and classification of a feature. This network is created by three layers: convolution layer, pooling layer, and finally, a fully connected layer.

Convolution Layer: This is the primary layer of a network which performs the feature extraction from the images. Here, patterns in images are recognized by the convolution of the feature which is automatically acquired by learning. Filters are utilized to obtain features from an image. Pooling Layer: This layer is used to

reduce the size of the feature-maps as to the response of the filter, such as sub-sampling, spatial-pooling, average-pooling, down-sampling, maximum-pooling, and sum-pooling. Maximum pooling layers are often applied in CNN architectures that calculate the maximum value in each patch of each feature map. For example, if there is 4×4 image and it is passed by max-pool, the image is converted to 2×2 pixel image. Fully Connected Layer: This layer is in charge of categorization. All extracted features created in convolution layer is given to this layer and it learns the features for categorization. This layer consists of several nodes called as neurons, and all neurons in a layer is attached to all other neurons in the next layer. Each layer consists of nonlinear activation functions like Sigmoid and ReLU. Convolutional neural networks are designed by assembling all the layers mentioned previously together.

3.4 Calculation of Accuracy

In order to calculate the accuracy of all the methods, various evaluation units are used and are given below:

$$\text{Accuracy} = (TP + TN)/(TP + TN + FP + FN)$$
$$\text{Sensitivity} = (TP)/(TP + FN)$$
$$\text{Precision} = (TP)/(TP + FP)$$

Here, true positive (TP) is the positive prediction of COVID-19 positive cases. False negative (FN) is the negative prediction of the positive cases. True negative (TN) is the negative prediction of the negative case. False-positive (FP) is the positive prediction of the negative case.

4   Summary of Survey

In this section, various techniques used by many research scholars were compared, examined and discussed. One such research was done by Wang et al. [4], where they created an open-access dataset known as COVIDX for training and evaluation of their system. COVIDX consists of 13,975 CXR images from 13,870 patients, which includes 358 images from 266 distinct COVID-19 patients. There were 8,066 patients who do not have pneumonia and 5,538 patients who have non-COVID-related pneumonia. The key motive of this study was to develop a manually specified design requirements network system integrated with machine operated inspection, to create a specific network architecture for identification of COVID-19 called as COVID-Net. COVID-Net is an open-source network design which utilizes lightweight residual PEPX (projection-expansion-projection-extension) architecture. It is a specialized

lightweight exploration approach that allows extensive representation capabilities while minimizing the computational complexity. In this study, it was observed that the COVID-Net was more sensitive than both ResNet50 and VGG19 for COVID-19-specific cases. Moreover, it was highly successful in the positive predictive value (PVP) for all COVID, non-COVID and normal categories as compared to the other architectures, with a highest of 98.9% of PVP for COVID-19 and had an accuracy of 93.3%. However, to further enhance the results in this study, some simple procedures can be integrated in this system such as data augmentation, which can be done to the dataset for improving the generalization of the model.

Another such research was done by Farooq et al. [6], where they used the residual rural network (ResNet) with 50 layers for the diagnosis of COVID-19. They also used the dataset COVIDX by Wang et al. [4] for training and evaluation. To further maximize the generalization of any model, they used the images after augmentation. Data augmentation also helps in increasing the size of the input training data and was performed by the use of FastAI framework. Here, training of the model was done in several stages with varying the size of the input image, which was done by gradual resizing which tends to enhance generalization further. In this study, a maximum accuracy of 96.23 was achieved by utilizing the COVID-ResNet architecture, while getting an excellent PVP of 100% for actual COVID-19 cases. However, in this research, it was observed that the accounted COVID-19 positive cases were less in comparison to non-COVID cases and the pneumonia cases. Therefore, more images of CXR COVID-19 positive patient are required for assembling a more robust system.

According to a study done by Alhudhaif et al. [12], bias problem was highlighted which was caused by databases. It was observed that some of the databases used by researchers contained loss in images, because the image features such as brightness, contrast and sharpness were changed by the researchers which was resulting in the bias problem. Therefore, considering this problem a generalized CNN design was developed to avert any biasing problem. In total, 1218 CXR images consisting of COVID-19 pneumonia and non-COVID pneumonia were assembled from different datasets including Wang [4], Kaggle, and Cohen [15]. This study primarily focused on binary classification of the cases using ResNet, DenseNet, and SqueezeNet architectures, which were pre-trained by ImageNet database then retrained by transfer learning method to create a CNN architecture. Stochastic gradient descent method was utilized for retraining some CNN models. Here, the activation mapping of CXR was achieved by gradient weighted class activation mapping technique to maximize the visibility by prioritizing critical infected areas of lungs. Fivefold cross-validation process was used to examine the binary classification results of the network architecture. Here, DenseNet-201 achieved the maximum accuracy of 94.96% for the COVID-19 positive pneumonia.

A model proposed by Hall et al. [5] consists of a model trained on 102 COVID and 102 non-COVID pneumonia images with a tenfold cross-validation method on CXR images utilizing pre-trained VGG-16 and Resnet50 neural network models. Here, validation method was utilized to approximate the feasibility of the CXR images for COVID-19 diagnosis. Dataset utilized was created by assembling from three different sources including Cohen, Radiopaedia, and SIRM. However, some

Table 2 Comparison of comparable studies for detection of COVID-19

Author	Dataset images	Dataset used	Model	Accuracy (%)
Wang	358 COVID-19, 5538 pneumonia, 8066 normal	COVIDx by Wang	COVID-Net	93.3
Hall	125 COVID-19, 208 pneumonia	Radiopaedia + SIRM + Cohen	ResNet50, VGG16	89.20
Farooq	68 COVID-19, 1203 normal, 1591 pneumonia	COVIDx	ResNet	96.23
Alhudhaif	368 COVID-19 pneumonia, 850 pneumonia	Kaggle + COVIDx + Cohen	DenseNet-201	94.96
Ghoshal	68 COVID-19, 1583 normal, 2786 bacterial + 1504 viral pneumonia	Cohen	Bayesian CNN	92.90
Horry	100 COVID-19, 100 pneumonia, 200 normal	Cohen + National Institute of Health	VGG19	83.00

loss in compression was observed in the images by the author. As they did not have a large dataset, they followed a snapshot ensemble technique of five classifiers. In the VGG network, there were 13 convolution layers, average pooling layer, fully connected layer followed by classification layer. The method recommended by them achieved an accuracy of 89.2% with 80.3% PVP, and they achieved 83.3 to 96% true positive rate. Some of the limitations observed by the author were the requirement of full-resolution CXR images in the dataset which would be more useful, and the dataset used in this study only include the patients with CXR observations, which result in incompetency to identify the illness in some patients who do not have any recognizable finding by the radiologist. Moreover, the amount of available COVID-19 CXR images were quite low as compared to other images in the dataset used. A comparison table of some related studies for the automated COVID-19 identification using CXR images is given in Table 2, comparison of the number of images is given in Fig. 2, and a comparison graph is given in Fig. 3.

5 Conclusions

The key motive of this study is to gain knowledge about how to create a well-functioning process that can help to aid the diagnosis and treatment procedure of COVID-19 through early detection. In this review, various models that can identify COVID-19 infections with the help of CXR images were accounted and they can detect COVID-19 with accuracy rates of above 90%. A maximum accuracy of 96.23
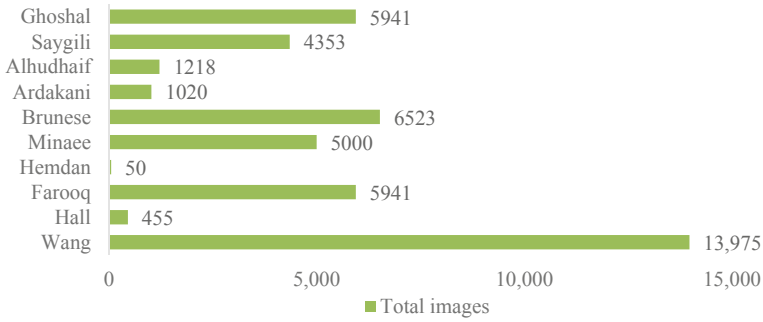
Fig. 2 Number of images comparison chart
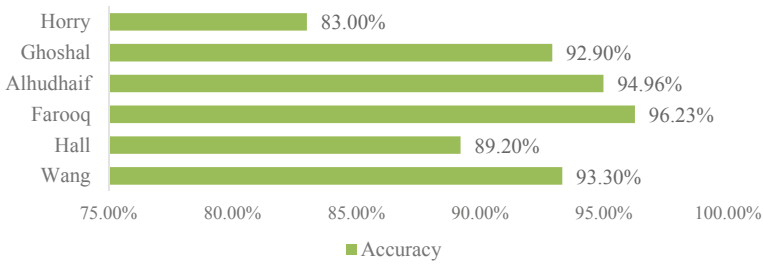
Fig. 3 Accuracy comparison chart

was achieved by utilizing CXR images for the identification of COVID-19 positive cases. It was noted that COVID-19 infections can be successfully identified with a high accuracy rate within a short period of time utilizing machine learning and image processing techniques. Thus, these types of system are highly applicable for medical use. In many models observed, various image processing methods were implemented on CXR in the preprocessing stage. Then, by implementing many classifiers to the preprocessed images, it was observed that the model attained a greater accuracy. It has been observed that Deep CNN, VGG, ResNet, DenseNet, SqueezeNet can identify COVID-19 positive cases very accurately. However, some conditions such as age and previous illnesses are problems that can alternate the degree of this illness. Therefore, the absence of impersonal information on the patients is a drawback of this system, and there is no data on which particular stage of the illness was the image taken from the patient. Some of the studies did not use full resolution X-ray images, which could be more useful for the analysis. Moreover, the size of the dataset matters a lot; therefore, a large dataset with many sample images in each group like COVID-19 or non-COVID-19 is still required.

References

1. WHO Coronavirus (COVID-19) Dashboard, World Health Organization, As of 30 Nov (2021)
2. World Health Organization, What to Know About COVID-19 Diagnosis, Available online: https://www.healthline.com/health/coronavirus-diagnosis (2021)
3. Kulkarni R, Patil HP, Palkar S, Lalwani S, Mishra AC, Arankalle V (2021) Anti-SARS-CoV-2 IgG antibody response among Indian COVID-19 patients using β-propiolactone-inactivated, whole virus-based indirect ELISA. J Virol Methods 287
4. Wang L, Lin ZQ, Wong A (2020) A tailored deep convolutional neural network design for detection of covid-19 cases from chest radiography images. J Netw Comput Appl
5. Hall LO, Paul R, Goldgof DB, Goldgof GM (2020) Finding covid-19 from chest x-rays using deep learning on a small dataset, preprint arXiv:2004
6. Farooq M, Hafeez A (2020) Covid-resnet: A deep learning framework for screening of covid19 from radiographs, preprint arXiv: 2003
7. Hemdan E-D, Shouman K (2020) Covidx-net: a framework of deep learning classifiers to diagnose covid-19 in x-ray images, preprint arXiv:2003
8. Minaee S, Kafieh R, Sonka M, Yazdani S, Soufi GJ (2020) Deep-covid: predicting covid-19 from chest x-ray images using deep transfer learning, arXiv preprint arXiv 2004
9. Brunese L, Mercaldo F, Reginelli A, Santone A (2020) Explainable deep learning for pulmonary disease and coronavirus COVID-19 detection from X-rays. Comput Methods Programs Biomed 105608
10. Horry MJ, Paul M, Ulhaq A, Pradhan B, Saha M, Shukla N (2020) X-Ray ımage based COVID-19 detection using pre-trained deep learning models (2020)
11. Ardakani AA, Kanafi AR, Acharya UR, Khadem N, Mohammadi A (2020) Application of deep learning technique to manage COVID-19 in routine clinical practice using CT images: Results of 10 convolutional neural networks. Comput Biol Med 121
12. Alhudhaif A, Polat BK, Karaman O (2021) Determination of COVID-19 pneumonia based on generalized convolutional neural network model from chest X-ray images. Expert Syst Appl 180
13. Saygili A (2021) A new approach for computer-aided detection of coronavirus (COVID-19) from CT and X-ray images using machine learning methods. Appl Soft Comput 105
14. Ghoshal B, Tucker A (2020) Estimating uncertainty and interpretability in deep learning for coronavirus (COVID-19) Detection. arXiv preprint arXiv:2003.10769
15. Cohen JP, Morrison P, Dao L (2020) COVID-19 image data collection. https://github.com/ieee8023/covid-chestxray-dataset

Design and Analysis of a Biconcave DRA by Using Machine Learning Algorithms for 5G Application

Ribhu Abhusan Panda and Archana Patnaik

Abstract   The conventional shape of dielectric resonating antenna has been modified to a biconcave shape having maximum arc-to-arc distance equal to the wavelength that has been calculated from the design frequency of 28 GHz. The height of the dielectric resonator has been selected as 3.8 mm with the material Alumina 92_pct having the dielectric constant 9.2. The dimension of the substrate has been taken as 40 mm × 40 mm × 1.6 mm. The broadband characteristic has been obtained with the −10 dB bandwidth more than 6 GHz. The frequency range has been considered from 22 to 28 GHz which includes many applications like satellite communication and 5G communication. Multiband characteristic has been obtained from the simulation result at the frequencies 22.8, 25 and 27.54 GHz. The simulation has been carried out by Ansys Electronic Suit-HFSS (high-frequency structure simulator). Optimization of the parameters have been done by using different machine learning algorithms, and a comparative study has been made.

Keywords Dielectric resonating antenna (DRA) · Biconcave · 5G · Multiband · Machine learning algorithm · S_{11}

1   Introduction

Earlier to 1980s, the uses of dielectric resonators (DRs) had been limited as filters and oscillators. However, the specific characteristics of DRs like, with dielectric constant more than 20, the Q-factor would be between 50 and 500, sometimes as large as 10,000, had not been exploited. Thus, this isolated the dielectric resonators as energy storing device. With advanced time, the DRAs have been evolved into different shapes and their applications to specific uses have been enunciated [1–5]. In recent years the

R. A. Panda
Department of Electronics and Communication Engineering, GIET University, Gunupur, Odisha, India
e-mail: ribhuabhusanpanda@giet.edu

A. Patnaik (✉)
Department of Computer Science Engineering, GIET University, Gunupur, Odisha, India
e-mail: archanapatnaik@giet.edu

conventional shape of patches has been modified as per the requirement to operate the antenna for specific applications [6–11]. Here, the proposed has been designed for 28 GHz which is suitable for 5G application. In past few years, machine learning algorithm played a vital role in optimizing the parameters. To optimize the parameters for the efficient design of proposed structure, the machine learning algorithms have been used, and a comparative study has been made for those algorithms. In order to evaluate the performance of the antenna, different types of regression are used. Based on the electromagnetic nature [12] of antenna, the resonance frequency lies within the range of 2.4–2.5 GHz. In case of microstrip patch antenna [13], support vector regression is used for finding best results where error rate is 3 dB which seems to be very less. Similarly, for double T-shaped monopole antenna [14], sparse linear regression is used as the evaluation parameter.

2   Design of the Proposed DRA

The proposed structure has basically three parts: substrate, ground pane and novel dielectric resonator. The dimension of the substrate has been considered as 40 mm × 40 mm × 1.6 mm. Ground plane has been designed with the dimension 40 mm × 40 mm × 1.6 mm. The novel biconcave DRA has the height equals to 3.8 mm. The minimum arc-to-arc length is considered as 10.71 mm which is the wavelength that has been determined from the design frequency 28 GHz. Aperture coupled feed has been used for the proposed structure. The design has been shown in Fig. 1. The value of the proposed arc-to-arc length has been determined by the following formula (Table 1).

$$f = \frac{c}{2\pi \sqrt{\varepsilon_r}} \frac{X^I mn}{r} \tag{1}$$

3   Optimization of Parameters Using Machine Learning Algorithm

Two prominent parameters associated with the dielectric resonator are arc-to-arc length and height of dielectric resonator. These have been optimized by using different machine learning algorithms like linear regression, logistic regression, least angle regression, lasso regression and ridge regression. In order to analyse maximum arc length, we have used a regression-based machine learning algorithm that includes linear regression, logistic regression, least angle regression, lasso regression and ridge regression. Based on the analysis, the performance is evaluated which is represented in the table below.
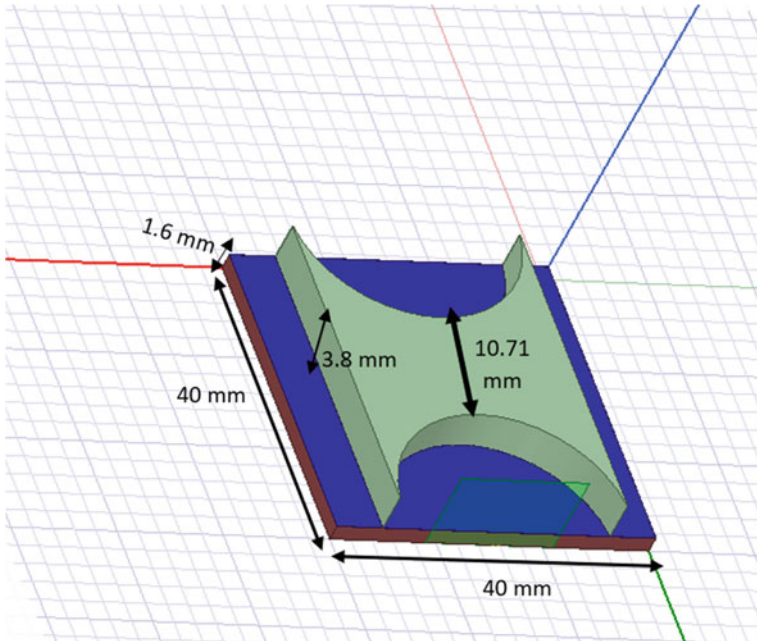
Fig. 1 Proposed biconcave DRA

Table 1 Parameters for the design of the proposed DRA

Design parameters	Symbol	Value
Frequency	f	28 GHz
Wavelength	λ	10.71 mm
Feed width	w_f	2 mm
Height of dielectric resonator	h	3.8 mm
Zeros of the derivatives of Bressel function	$X^I mn$	11.77
Dielectric constant of substrate	ε_r	4.4

3.1 Linear Regression

It is a supervised learning-based statistical method used for analysis of continuous data. In our work, we have considered various data point and estimated the relationship between the minimum arc which is considered as target variable and height of DRA which is considered as the predictor variable (Fig. 2).

In the above figure, the minimum arc-to-arc length is evaluated by using linear regression. It is observed that the best arc length is 10.5 mm where height of DRA

Fig. 2 Minimum arc length by using linear regression
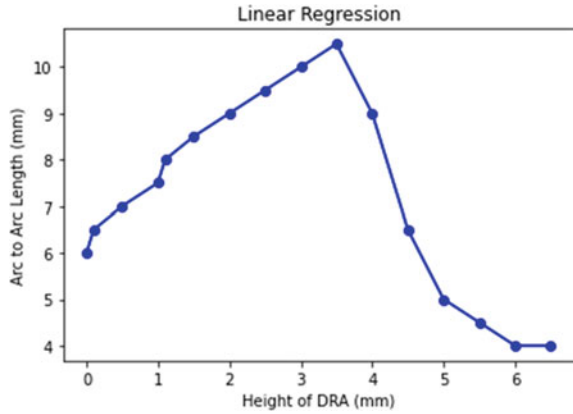
is equal to 3.5 mm. The value plotted in the graph lies within the range of 10.3–10.5 mm. After achieving the minimum value, the arc length is reduced by some point, and finally, it reached 4.0 mm.

3.2 Logistic Regression

It is predictive-based algorithms which work on categorical value of data. It is based on the concept of regression line by considering the minimum likelihood function that predicts minimum value between 0 and 1. In our work, we are using this algorithm in order to evaluate maximum threshold arc length of the E-shaped antenna (Fig. 3).

In the above figure, the minimum arc-to-arc length is evaluated by using logistic regression. It is observed that the best arc length is 10.0 mm where height of DRA

Fig. 3 Minimum arc length by using logistic regression
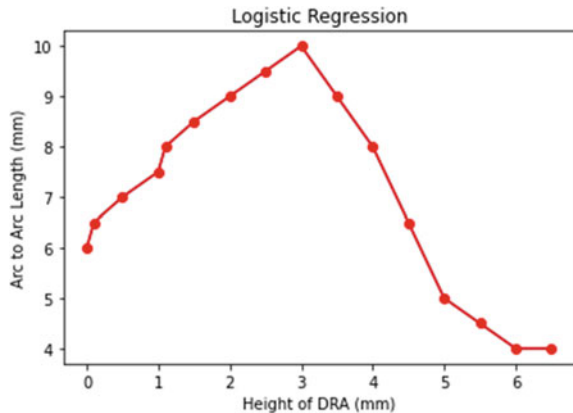
Fig. 4 Minimum arc length by using least angle regression
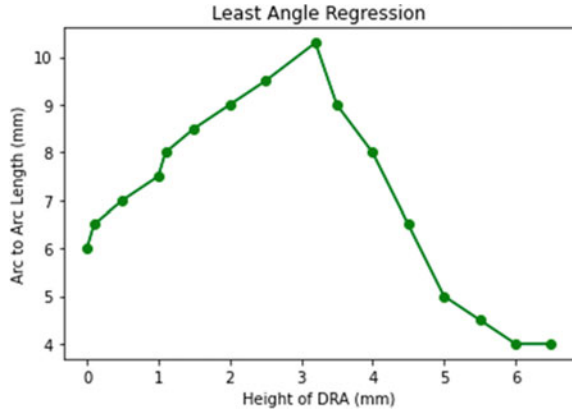
is equal to 3.0 mm. After achieving the minimum value, the arc length is reduced by some point, and finally, arc length reaches 4.1 mm at height of 6.0 mm.

3.3 Least Angle Regression

It is a type of regression algorithm that fits best for higher dimensional data value. It considers the average attributes correlated value and process the data in forward stepwise regression methods by normalizing all the data values. Here, the height of DRA acts as a predictor and arc length as the target estimator at equiangular distance (Fig. 4).

In the above figure, the minimum arc-to-arc length is evaluated by using least angle regression (LAR). It is observed that the best arc length is 10.3 mm where height of DRA is equal to 3.5 mm. After achieving the minimum value, the arc length is reduced by some point, and finally, arc length reaches 4.3 mm at height of 6.1 mm.

3.4 Lasso Regression

Least **A**bsolute **S**hrinkage and **S**election **O**perator (LASSO) regression is a statistical analysis of data model which is regularized by considering the relevant features. In our work, L1 regularization is done by variable parametric selection. L2 regularization is also done by considering height and arc length of the antenna (Fig. 5).

In the above figure, the minimum arc-to-arc length is evaluated by using lasso regression. It is observed that the best arc length is 10.7 mm where height of DRA is equal to 3.7 mm. After achieving the minimum value, the arc length is reduced by some point, and finally, arc length reaches 4.0 mm at height of 6.5 mm.

Fig. 5 Minimum arc length by using lasso regression
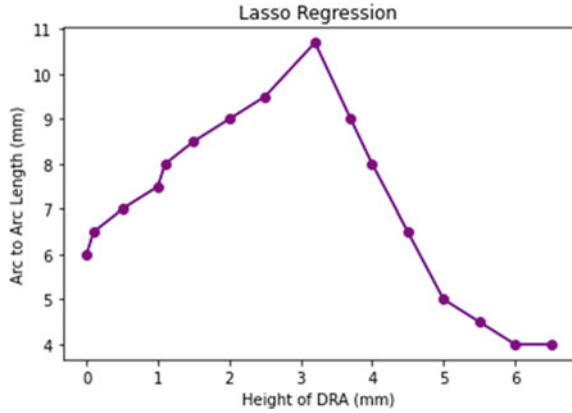
3.5 Ridge Regression

It is machine learning regressor which is used to analyse widely affected data by the use of multi-collinearity. In order to reduce the bias and variance of the source data set, we have used the regularization concept and done the comparison between the actual and predicted value of the arc length which is highly correlated with each other (Fig. 6).

In the above figure, the minimum arc-to-arc length is evaluated by using ridge regression. It is observed that the best arc length is 9.9 mm where height of DRA is equal to 3.2 mm. After achieving the minimum value, the arc length is reduced by some point, and finally, arc length reaches 4.3 mm at height of 6.4 mm.

Fig. 6 Minimum arc length by using ridge regression
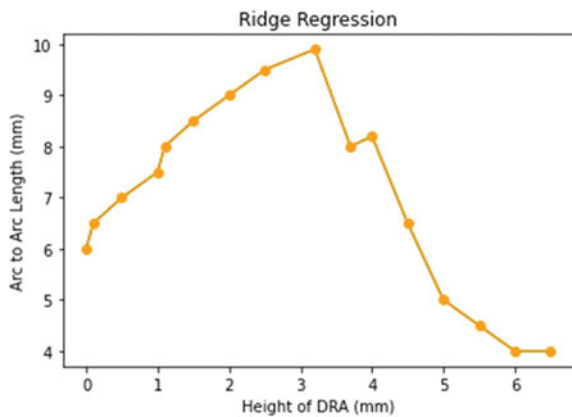
3.6 Comparison of Different Machine Learning Regressor

In Table 2, accuracy is predicted for all the five regressors used for performance evaluation of the antenna (Fig. 7).

In the above figure the performance of each machine learning regressor is evaluated by using five regression algorithms where x-axis indicates the maximum arc length and the y-axis indicates accuracy In simple linear regression, logistic regression, least angle regression, lasso regression and ridge regression, the value of maximum arc-to-arc length is 10.5 mm, 10.0 mm, 10.3 mm, 10.7 mm and 9.9 mm, respectively. It is observed that the best value is obtained in lasso regression which is equal to 10.71 mm. It is observed that accuracy rate is more in case of lasso regression as compared to other types of regressors used. This algorithm shrinks the regression coefficient and helps to improve the quality of the target estimator resulting

Table 2 Performance evaluation using accuracy parameters

Sl. No	Regressor name	Maximum Arc-to Arc length (mm)	Accuracy (%)
1	Linear regression	10.5	90
2	Logistic regression	10.0	92
3	Least angle regression	10.3	89
4	Lasso regression	10.7	99
5	Ridge regression	9.9	95
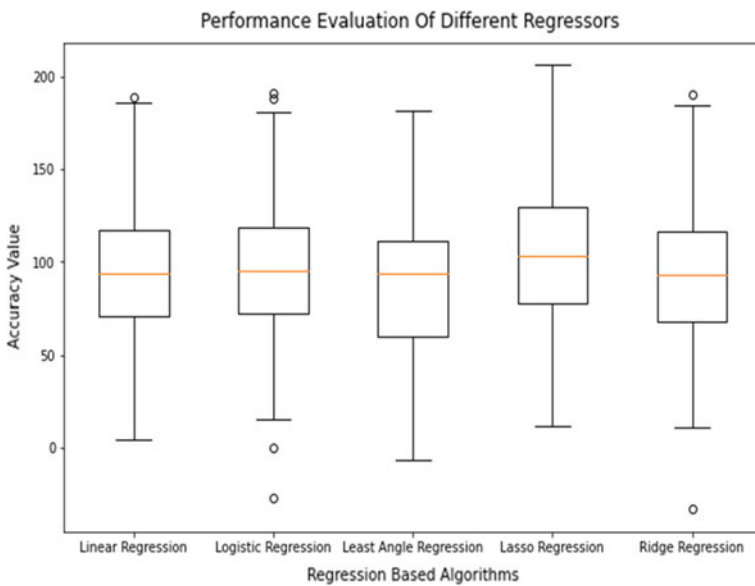
Fig. 7 Comparison of different machine learning algorithms
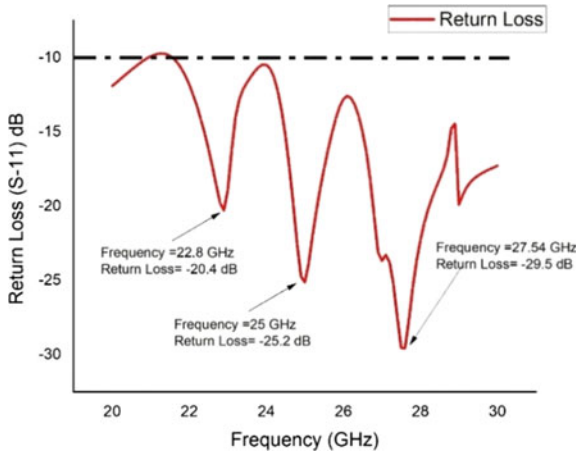
Fig. 8 S-parameter

in maximum arc-to-arc length value. That is why lasso regression is considered to be the best algorithm

4 Result from the Simulation

From the S-Parameter plot which has been shown in Fig. 8, it has been observed that multiple resonant frequency has been observed at 22.8, 25 and 27.54 GHz with return loss −20.4 dB, −25.2 dB and −29.5 dB, respectively. The antenna gain for the proposed structure has been determined with respect to frequency, and the peak antenna gain has been found out to be 10.5 dB which has been illustrated in Fig. 9.

The radiation pattern is symmetric around the structure. The 3D radiation pattern has been shown in Fig. 10. The field vectors of electric field have been shown in Fig. 11. Table 3 includes the simulated parameters, and Table 4 includes comparison of proposed work with other recent works.

5 Conclusion

The proposed DRA has an efficient performance regarding resonant frequency (27.8 GHz) and antenna gain (10.5 dB). The uniqueness of the antenna is the shape of the dielectric resonator which is an unconventional shape. Machine learning algorithms have been implemented to optimize the design parameters to obtain best results for 28 GHz applications that include 5G communication.
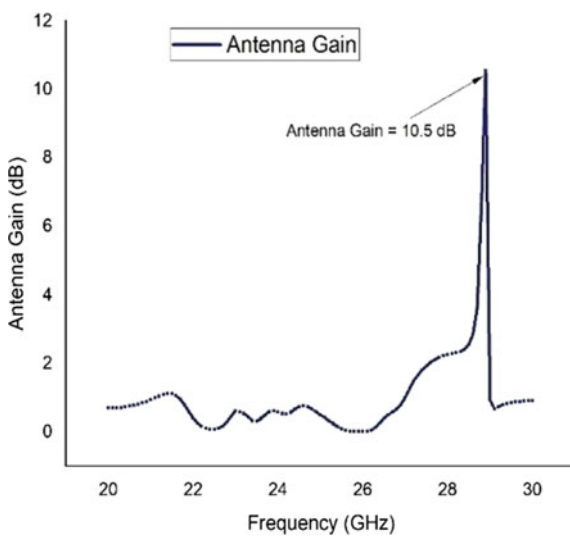
Fig. 9 Gain versus frequency
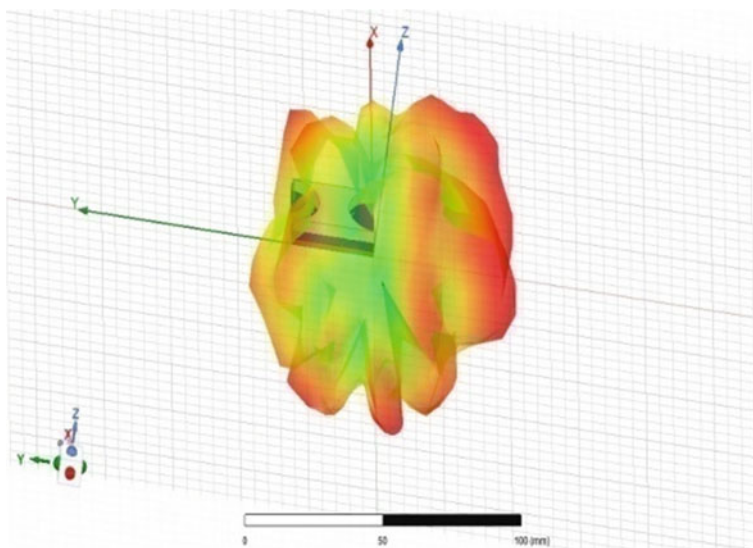
Fig. 10 3D radiation pattern
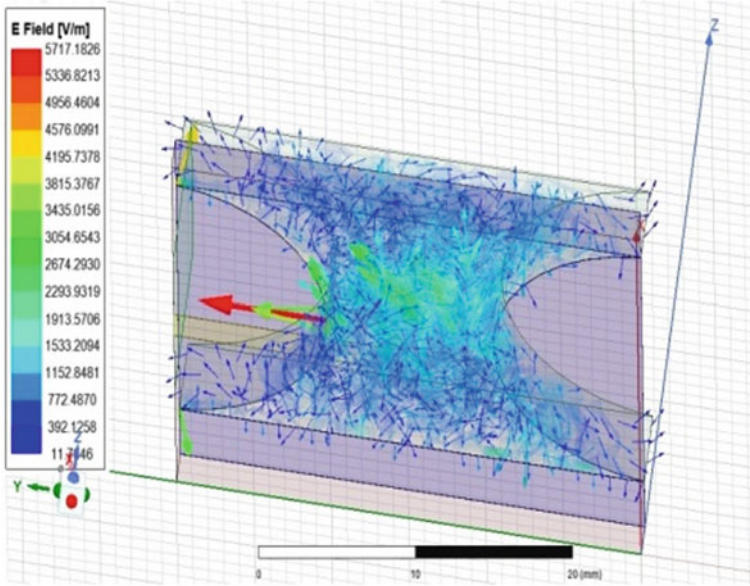
Fig. 11 E-field vectors

Table 3 Parameters from the simulation

Parameter	Return loss	Resonant frequency	Peak gain	Peak directivity	Radiation efficiency	Bandwidth
Value	−29.5 dB	27.54 GHz	10.5 dB	11.28 dB	93%	10 GHz

Table 4 Comparison of works

Reference	DRA type	Shape	Dimension (Radius, Height) (mm)	Resonant frequency (GHz)	Application of machine learning algorithm
[1]	Filter antenna	cylindrical (Conventional)	6.75, 9	5.5	NO
[3]	Aperture coupled	Cylindrical (Conventional)	4.3, 5.1	10.5	NO
[4]	Compact polarized	Ring (Conventional)	11.5, 5	12.38	NO
Proposed	Aperture coupled	Biconcave (Unconventional)	10.71, 9.1	27.54	YES

References

1. Mallat NK, Iqbal A (2021) Out-of-band suppressed SIW-DRA based filter-antenna subsystem with flexible bandwidth and transmission zeros. AEU-Int J Electron Commun 135:153735

2. Meher PR, Behera BR, Mishra SK (2021) A compact circularly polarized cubic DRA with unit-step feed for Bluetooth/ISM/Wi-Fi/Wi-MAX applications. AEU Int J Electron Commun 128:153521. https://doi.org/10.1016/j.aeue.2020.153521. ISSN 1434-8411

3. Ibrahim MS, Attia H, Cheng Q, Mahmoud A (2020) Wideband circularly polarized aperture coupled DRA array with sequential-phase feed at X-band. Alexandria Eng J 59(6):4901–4908. ISSN 1110–0168

4. Meher PR, Behera BR, Mishra SK, Althuwayb AA (2021) Design and analysis of a compact circularly polarized DRA for off-body communications. AEU Int J Electron Commun 138:153880. https://doi.org/10.1016/j.aeue.2021.153880. ISSN 1434-8411

5. Kethavathu SN, Singam A, Muthusamy P (2021) Compact symmetrical slot coupled linearly polarized two/four/eight element MIMO bowtie DRA for WLAN applications. AEU Int J Electron Commun 135:153729. https://doi.org/10.1016/j.aeue.2021.153729. ISSN 1434-8411

6. Alali SAI, Sabouni F, Kishk AA (2019) Thin perforated A-shaped DRA for UWB applications. In: 2019 49th European microwave conference (EuMC), pp 69–72. https://doi.org/10.23919/EuMC.2019.8910884

7. Meher PR, Behera BR, Mishra SK (2018) Design of different shaped DRAs for 60 GHz millimeter-wave applications. IEEE Indian Conf Antennas Propog (InCAP) 2018:1–4. https://doi.org/10.1109/INCAP.2018.8770785

8. Panda RA, Mishra D, Kumari P, Hansdah B, Singh RK (2020) Gauging trowel shaped patch with circular rings for 60 GHz WLAN. In: 2020 7th international conference on signal processing and integrated networks (SPIN), pp 113–116. https://doi.org/10.1109/SPIN48934.2020.9070868

9. Panda RA, Mishra D (2020) Efficient design of Bi-circular patch antenna for 5G communication with mathematical calculations for resonant frequencies. Wirel Pers Commun 112:717–727. https://doi.org/10.1007/s11277-020-07069-9

10. Panda RA, Kumari P, Naik J, Negi P, Mishra D (2021) Flower shaped patch with circular defective ground structure for 15 GHz application. In: Abraham A, Panda M, Pradhan S, Garcia-Hernandez L, Ma K (eds) Innovations in bio-inspired computing and applications. IBICA 2019. Advances in intelligent systems and computing, vol 1180. Springer, Cham. https://doi.org/10.1007/978-3-030-49339-4_24

11. Panda RA, Mishra D, Panda EP, Patnaik N (2020) Reshaped circular patch antenna with optimized circular slot for 5G application. In: Hitendra Sarma T, Sankar V, Shaik R (eds) Emerging trends in electrical, communications, and information technologies. Lecture notes in electrical engineering, vol 569. Springer, Singapore. https://doi.org/10.1007/978-981-13-8942-9_63.

12. Abgaryan G, Tumakov D (2018) Designing a koch-type wire antenna by regression analysis. In: 2018 IEEE east-west design and test symposium (EWDTS). IEEE, pp 1–4

13. Ülker S (2019) Support vector regression analysis for the design of feed in a rectangular patch antenna. In: 2019 3rd international symposium on multidisciplinary studies and innovative technologies (ISMSIT). IEEE, pp 1–3

14. Sharma Y, Wu J, Xin H, Zhang HH (2017) Sparse linear regression for optimizing design parameters of double T-shaped monopole antennas. In: 2017 IEEE international symposium on antennas and propagation and USNC/URSI national radio science meeting. IEEE, pp 347–348

An Approach for the Estimation of Rotor Position of PV Fed Switched Reluctance Motor Using ANFIS

Adunuri Srilekha, Kannan Kaliappan, and Ranjith Kumar

Abstract In this work to eliminate torque ripple, a solar standalone photovoltaic (PV) powered fed system (SRM) switched reluctance motor is controlled using an ANFIS-based torque controller. The major draw drawbacks of the SRM are high torque ripple and mechanical vibration, as well as acoustic noise. This is due to its nonlinearity and double salient structure, which causes it to produce torque ripple. In switched reluctance motors, there isn't a single developed solution for minimizing torque ripples. Here in the (MPPT) maximum power point track technique (P&O) Perturb and Observe algorithm is enforced to the Photo Voltaic (PV) system to draw out the highest level of power from the Photovoltaic panel. Modeling and analysis of a switching reluctance motor are performed in this study. Comparisons between ANFIS and other techniques show that it is more accurate, less complex, stable, and generalizable than other techniques. The SRM drive is driven by electronic commutation, which is attained by a position hall sensor and an encoder, discretely. Adaptive neuro-fuzzy inference (ANFIS) is put forward in this paper as a speed and current control approach for the SRM motor to reach the target of motor velocity as mentioned in the reference velocity. The simulation model is created in MATLAB/SIMULINK and results are analyzed with ANFIS and FUZZY.

Keywords (SRM) switched reluctance motor · (ANFIS) Adaptive neuro-fuzzy ınference system · Perturb and observe (P&O) algorithm · Photo voltaic(PV) array

1 Introduction

Recently, variable-speed drives are using SRMs in industrial and other applications are gradually increasing [1, 2]. Giant voltage and current switches for SRM control are now available at a reasonable price in comparison to permanent magnet brushless and induction motors [1], making that feasible to take advantage of the SRM's uncomplicated design, stiffness, and low production cost. The most fundamental

A. Srilekha (✉) · K. Kaliappan · R. Kumar
Electrical and Electronics Engineering, Sreenidhi Institute of Science and Technology, Hyderabad, India
e-mail: adunurisrilekha@gmail.com

property of the SRM would be a nonlinear relationship between phase current and position of the rotor that makes up to achieve great efficiency, the SRM runs at higher saturation levels of the magnetic circuit. Due to this, the SRM's mapping between input and output variables, as well as its parameters, is extremely nonlinear [2]. There has been a lot of development in the studies on torque ripple and acoustic noise reduction. Magnetic saturation causes phase inductance to vary with motor current, resulting in substantial errors in both instantaneous and average torque [3, 4]. High performance is dependent on minimizing torque ripple [5]. When it comes to optimizing the arrangement of SRMs, the properties of the magnetic field play a key role. As a result of machine design or their control circuitry, SRM torque ripples can be eliminated. Finite-element programs have been used by several researchers to develop motors and determine their torque, current, and rotational angle properties. [6, 7]. As the torque and speed are inversely proportional, it is necessary to eliminate ripples to minimize reluctance [8]. Direct instantaneous torque control (DITC) technology was used to lessen torque ripple, but it had a restricted operating range. As speed increases, a new controller must be used. As a result of torque ripple minimization, TSF (torque sharing function) and current control techniques were used, but these methods ignore mutual torque, which has a major impact in some applications [9]. It was proposed that switched reluctance motors be controlled by a single controller for wide-speed operations High-speed operation is carried out in single pulse mode by the suggested controller, which has a minimum output torque ripple at the low and medium speed [10]. In [11], torque ripple and flux are managed within the hysteresis limit by utilizing the direct torque control (DTC) approach. Here in the proposed method, SRM is connected to the PV system because, failure of the electrical grid has a significant impact on the economic growth of the country, as well as the industrial sector's development.

Renewable energy is on the rise, and grid dependence has declined in recent years. Due to the ease of installation of PV panels, growth in the count of solar PV rooftop systems in every residence minimizes issues of power quality and thereby increases power returned to the grid. In the past, a DC-DC boost converter has been utilized to augment the input power of PV power generation. Many converters' techniques have been studied in the literature, and every technique has its limitations to the same degree efficiency, dependability, and operating mode of the system, which have been examined. Using the MPPT control technology efficiency and makes it difficult for wind and solar energy systems to have a significant contribution. Using a capacitive converter, the suggested boost converter decreases threshold voltages, reduces inductor size, and also manages stability to give a constant dc voltage. In this paper, the mathematical calculations of the SRM control parameters for a given torque, command/rotor speed (working point), and bus voltage are analyzed. As the torque ripple increases in speed, it becomes more sensitive to the current profile. Based on its simple, easy-to-implement, and rapid dynamics, ANFIS- based Direct Torque Control is put forward in this study to lessen torque ripple at low speeds.

2 Mathematical Analyses of PV System

Due to their cost-effectiveness and ease of maintenance, P-I characteristics of photo-voltaic array inverter topologies are commonly employed in PV solar power generation systems. A significant number of PV diodes are connected to in S-P arrangement. The output current of the PV panel can be indicated as [22] (Fig. 1):

$$I = N_{\text{PP}}[I_{\text{PV}} - I_o(I_P - 2)] - \left(\frac{V + IR_S\tau}{R_p\tau}\right) \tag{1}$$

where

$$I = \exp\left(\frac{V + IR_s\tau}{V_TN_{SS}}\right) + \exp\left(\frac{V + IR_s\tau}{(P-1)V_TN_{SS}}\right) \tag{2}$$

$$\tau = \frac{N_{SS}}{N_{\text{PP}}} \tag{3}$$

V and I are corresponding, solar cell's output voltage and current Vt stands for the voltage of PV arrays, whereas Io the inverse saturation current IPV is the photocurrent and of PV arrays. Series resistance is equal to RS, while a parallel resistance equals RP Photovoltaic (PV) cell production is highly connected to solar irradiation. The PV array has high nonlinear VI characteristics when solar irradiance fluctuates. As it does not have a sustained amount of voltage nor a sustained current, it cannot supply a constant amount of electricity to a given load Most of the operating voltage range has a steady output current, but toward the open-circuit voltage the current declines rapidly. From the figure, it can be seen that the photovoltaic array output characteristics vary considerably under the impact of solar irradiance. The output power keeps increasing as solar irradiance increases.

Fig. 1 I-V and P-V characteristics of the PV system
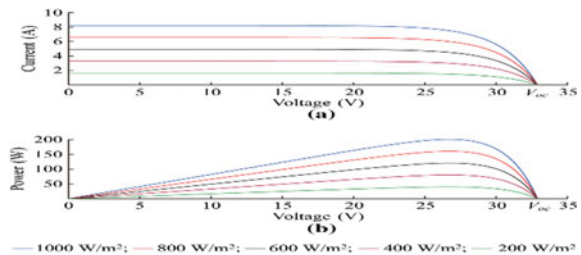
3 Proposed P&O Algorithm

Method of Perturb and Observe is a popular method. This approach uses minimum sensors. In this method, the operating voltage is tested, and the operating voltage is modified in a specified direction using an algorithm, and therefore the $\frac{d_P}{d_v}$ (Fig. 2).

If $\frac{d_P}{d_v}$ Pd v is sampled. If $\frac{d_P}{d_v}$ pd v is positive, the algorithm increases the voltage value toward MPP until $\frac{d_P}{d_v}$ pd v is negative. This process is imitated until the algorithm reaches MPP. This approach is unfitted when there is a considerable change in solar irradiation. The voltage fluctuates about the maximum power point (MPP) and never achieves a precise value.

For MPPT, this technique utilizes the instantaneous conductance $\frac{d_P}{d_v}$ and the P and O $\frac{d_P}{d_v}$. 1The location of the PV module's operating point in the P–V curve is decided based on the relationship between both values, as written in (4)–(6), i.e., (4) stipulated the PV module operating at the MPP, wherein (5) and (6) specifies the PV module performing on the left side and right side of the MPP in the P–V curve, individually. Perturb and Observe method is widely used.

$$\frac{d_p}{d_v} = -\frac{P}{V} \tag{4}$$

$$\frac{d_p}{d_v} > -\frac{P}{V} \tag{5}$$

$$\frac{d_p}{d_v} < -\frac{P}{V} \tag{6}$$

The equations given above are derived from the idea that at MPP the decline in the P–V curve is equivalent to zero

$$\frac{d_p}{d_v} = 0 \tag{7}$$

By revising (7), the below equation is obtained:

Fig. 2 Flow chart of perturb and observe
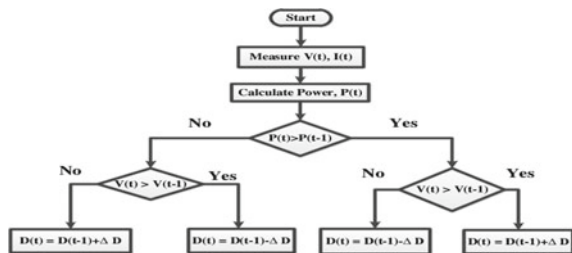
Fig. 3 Photovoltaic system
with the boost converter
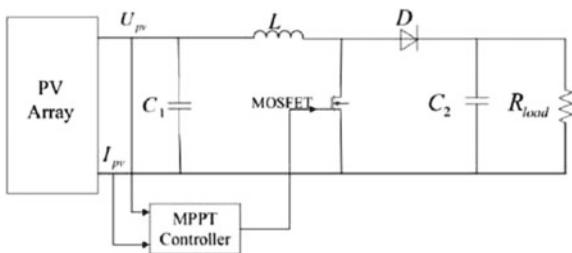
$$I + V \frac{d_i}{d_v} = 0 \tag{8}$$

If (5) is gratified, the duty cycle of the controller is required to be reduced, and vice versa. If (6) is gratified, in so much no change on the duty cycle (Fig. 3).

4 Principle of SRM

Changes in irradiance will affect current and voltage levels. This program takes as inputs PV module current and voltage fluctuations according to MPP, if the system is moving in the direction of positive ($dv > 0$), the duty cycle will proceed to move in the way of the previous step's disturbance, wherein it will carry on to move in the opposite way (dv 0). Alike, while the system works on the right-hand side of the MPP, the duty cycle will proceed to flow in the opposite direction of the previous step's disturbance for positive system operating direction ($dv > 0$), and in the opposite way of the previous step's disturbance for negative system operating direction (dv 0). Because the algorithm can correctly and appropriately determine the direction of the disturbance, it can solve the problem of system inaccurate results in the old technique.

$$V_j = Ri_j + \frac{d\lambda_j(\theta, i_j)}{dt} + \frac{d\lambda_i}{dt} \tag{9}$$

where Vj, be the jth phases winding voltage, ij for jth the phase current, j for the linking flux, R gives the phase winding's ohmic resistance, and h for the leaking flux. The relation between adjacent windings has been ignored (Fig. 4).

$$V_j = Ri_j + \frac{d\lambda_j(\theta, i_j)}{dt} \cdot \frac{di_j}{dt} + \frac{d\lambda_j(\theta, i_j)}{d\theta} \cdot \frac{d\theta}{dt} + \frac{d\lambda_i}{dt} \tag{10}$$

where, the increase in inductance (Linc) and the back EMF coefficient (Cw), both of which are inferior on current as well as the rotor angular position.

Fig. 4 8/4 switched reluctance motor geometric diagram
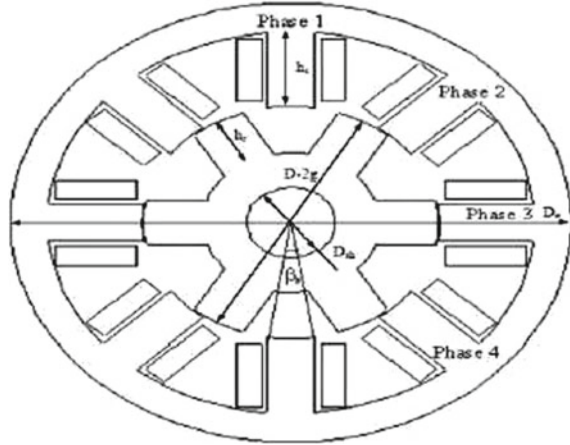
Considering L_k as the flux leakage:

$$\frac{d\lambda_j}{dt} = L_k \frac{di_j}{dt} \tag{11}$$

The mechanical equation of the rotor was recorded by the function of the electromagnetic Torque. Here the developed torque

$$T_e = \frac{1}{2} i^2 \frac{dL(\theta, i_j)}{d\theta} \tag{12}$$

By simplifying the magnetic linearity, the total generated torque is

$$T_T = \sum \frac{1}{2} i^2 \frac{dL(\theta, i_j)}{d\theta} \tag{13}$$

where T is the torque, θ is position of rotor angle, L gives the inductance of motor.

5 Arrangement of Feedback Controller

The feedback controller generates a controlled variable by comparing the regulated actual speed to the reference speed in order to impact system performance. To drives the SRM motor as the actuator the (ANFIS) Adaptive Neuro-Fuzzy Inference System is taken. The modeling of fuzzy and ANFIS controllers is covered in this section.
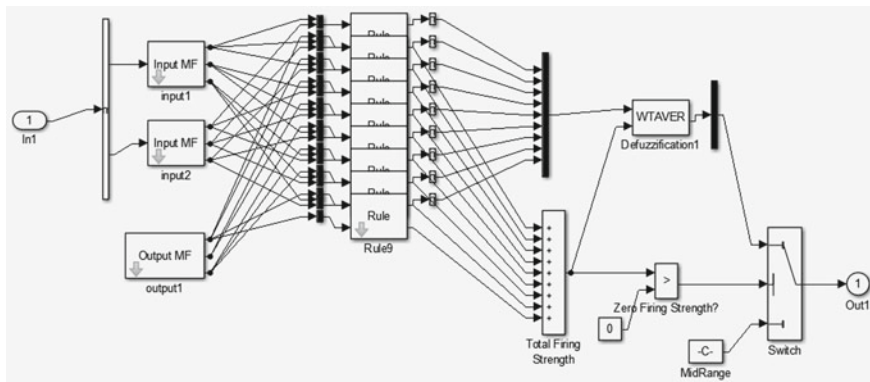
Fig. 5 ANFIS controller block diagram

A.   ANFIS Controller

In terms of efficiency and error rate, ANFIS, as a hybrid network, has the merits of both a neural and a fuzzy network. As a result, faults in the training network are propagated backward. For the ANFIS network's training rules, there are two types of input and output data. The command executes its tasks using five layers of state for training purposes. When training data is used to create a fuzzy inference system, the parameters of membership function were modulated using either back propagation algorithm, or a recursive least squares type approach in combination with back propagation alone. The learning process is accelerated when compared to using the gradient approach alone, which has an antipathy to get stuck in local minima when used alone. A common rule set having two fuzzy rules those are of If–Then in Takagi and Sugeno's types as shown below (Fig. 5).

$$\text{Rule1 : If } x \text{ is } A_1 \text{ and } y \text{ is } B_1 \text{ then } f_1 = p_1 x + q_1 y + r_1$$
$$\text{Rule2 : If } x \text{ is } A_2 \text{ and } y \text{ is } B_2 \text{ then } f_2 = p_2 x + q_2 y + r_2$$

The node function of the same layer belong to the same function family, which is detailed further below:

Layer 1: Each input node is fuzzified by associating with membership function of linguistic label, and thus the output layer specifies the degree of input. The obtained parameters in this layer are called premise/antecedent parameters.

Layer 2: This node is investigated as a "Rule node," and is symbolized by a letter T. It computes the linked rule's firing strength by using each node. The neuron's output can be calculated as follows:

$$\alpha_1 = L_1 a_1 * L_2 a_2 * L_3 a_3 \tag{14}$$

$$\alpha_2 = H_1 a_1 * H_2 a_2 * L_3 a_3 \tag{15}$$

$$\alpha_3 = H_1 a_1 * H_2 a_2 * H_3 a_3 \tag{16}$$

Layer 3: The normalization process takes place at each and every node, designated by a letter N, and the output is written as

$$\beta_1 = \frac{\alpha_1}{\alpha_1 + \alpha_2 + \alpha_3} \tag{17}$$

$$\beta_2 = \frac{\alpha_2}{\alpha_1 + \alpha_2 + \alpha_3} \tag{18}$$

$$\beta_3 = \frac{\alpha_3}{\alpha_1 + \alpha_2 + \alpha_3} \tag{19}$$

Layer 4: This layer is known as the layer of defuzzification, and it is often classified as the "then" section. Its outcome is the result of the normalization level and the individual rule output.

$$\beta_1 Z_1 = \beta_1 V B^{-1} . \alpha_1 \tag{20}$$

$$\beta_2 Z_2 = \beta_2 V B^{-1} . \alpha_2 \tag{21}$$

$$\beta_3 Z_3 = \beta_3 S^{-1} . \alpha_3 \tag{22}$$

Layer 5: The system's overall output (Z_0) is computed in this single node as summation of all the incoming signals and expressed as:

$$Z_0 = \beta_1 Z_1 + \beta_2 Z_2 + \beta_3 Z_3 \tag{23}$$

Hence, the vector input is normally sent through the layer by layer in the network, taking into account how the ANFIS learns the antecedent and conclusion parameters for the membership functions and rule set (Fig. 6).

The count of Membership functions for input vector variables e and ce is 3 * 3, respectively. The number of rules then 9 (3 * 3 = 9). The triangular membership function is used for two input variables (Fig. 7; Table 1).
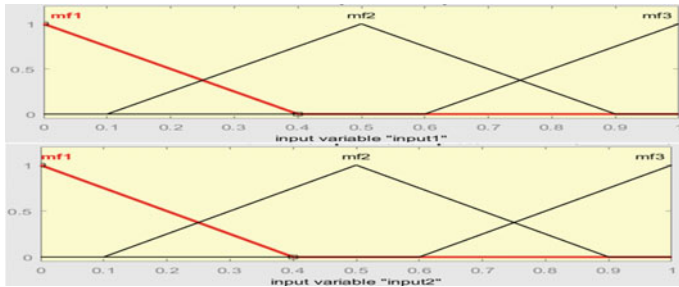
Fig. 6 Membership function of e and ce after trained
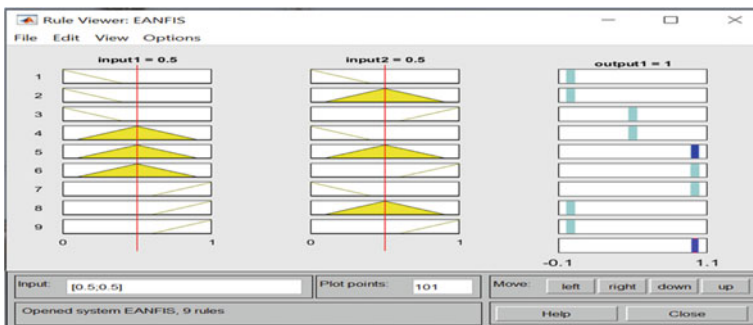
Fig. 7 FIS rule viewer

Table 1 Rules set for speed control

E/CE	Negative (N)	Zero (Z)	Positive (P)
Negative (N)	Negative (N)	Negative (N)	Negative(N)
Zero (Z)	Zero (Z)	Zero (Z)	Positive (P)
Positive (P)	Positive (P)	Positive (P)	Positive (P)

6   Simulation Results

Solar PV unit creates greatest voltage for 1000 W/m^2 irradiance measure inserted to refitted boost converter supplies constant input to SRM motor. The maximum power from the PV system is traced using the P&O MPPT method. The output is sent to the inverter, and the temperature is adjusted at $25°$C. The PV was irradiated at 1000 w/m^2, and the matching model and output voltage are given in Fig. 8. To authenticate the stability of the invented technique, the PV MPPT performance is varied by adjusting the PV irradiance level. The simulation is based on an 8/6 switching reluctance motor and is controlled by an ANFIS controller. The proficiency of the ANFIS controller method in the SRM is modeled using the MATLAB/SIMULINK module. To get

(a)

(b)

(c)

(d)
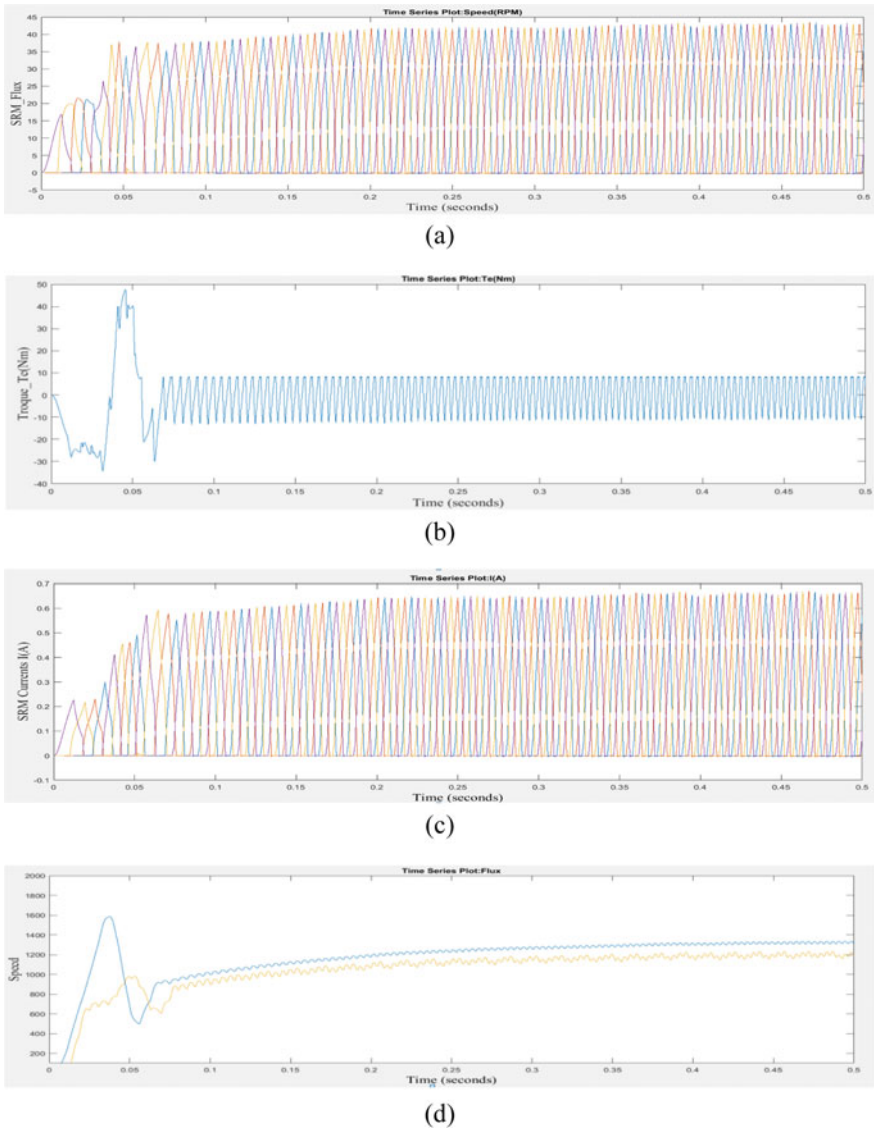
Fig. 8 **a** Flux, **b** torque, **c** currents, **d** speed wave form of the SRM

simulated the output performance of the proposed technique actuation system, an ANFIS controller is fetched to the solar powered SRM.

In the above Fig. 8d at time $t = 0.05$ the speed of the SRM motor expands from 1200 to 1500 and gradually moves to 2000. In the above Fig. 8b the Torque of the SRM motor is stable at the 4 at time $t = 0.05$ s. The SIMULINK modeling of the (SRM) switched reluctance motor connected to control system of ANFIS was

designed as an output result. Figure 8a–d demonstrate the performance of the flux, current in every phase, motor speed, and output torque characteristics. Numerous test outputs are produced utilizing the ANFIS control, and the proficiency of speed and torque variation with regard to time is depicted in Fig. 8. The testimonial speed parameter of the motor is determined in simulation, and the real speed parameter of the motor is collected from the motor during start-up SRM withdraws current and is restricted to the maximum value. The real and testimonial speeds are differentiated and difference/error calculated. When an error takes place, the Neuro-Fuzzy controller; similarly, the real and testimonial speeds of the motor are identical, indicating that there is no error. When the motor speed approaches its reference value, the beginning current peaks, and its peak value decreases.

7   Conclusion

In MATLAB/SIMULINK, an ANFIS controller-driven SRM motor drive was modeled and tested. The power of a solar PV system is changed by the boost converter for MPPT by the P&O algorithm. In order to match reference speed tracking, simulation is performed, and the proficiency of the proposed technique is tried out using ANFIS controller trained data. The Electronic commutation which has lesser switching losses is used to supply the DC-link voltage to SRM drive feeds via a modified inverter. The recommended converter is compact, has a greater power density, and provides a consistent output of PV voltage for the SRM motor actuation system. The ANFIS controller reduces SRM torque ripples, which improves motor drive efficiency and reduces rise in time and error percent. The simulation results illustrate the PV-tied SRM motor properties in conjunction with the ANFIS controller.

References

1. Mangu B, Akshatha S, Suryanarayana D, Fernandes BG (2016) Grid-connected PV wind-battery-based multi-input transformer-coupled bidirectional DC-DC converter for household applications. IEEE 4(3):1086–1095
2. Singh B, Mishra AK, Kumar R (2016) Solar powered water pumping system employing switched reluctance motor drive. IEEE Trans Ind Appl 99:1–1
3. Peng F, Ye J, Emadi A (2016) A digital PWM current controller for switched reluctance motor drives. IEEE Trans Power Electron 31(10):7087–7098
4. Hang J, Hung Y-R, Li L (2010) Switched reluctance motor direct torque control research based on RBF neural network and fuzzy adaptive PID controller. In: Second international conference on intelligent human-machine system and cybernetics
5. Tekwani PN, Umakant PN (2017) Analysis of carrier offset technique used in a five- level inverter scheme with emphasis on dc-link capacitor voltage balancing. In: Proceeding of the 6th Nirma university international conference on engineering (NUiCONE 2017), 23–25 November 2017, Nirma University, Ahmedabad, India, pp 1–9, Paper id 128

6. Ali Akcayol M (2004) Application of adaptive neurofuzzy controller for SRM. Adv Eng Softw
7. Divandari M, Rezaie B, Askari-Ziarati B (2016) Torque estimation of sensor less SRM drive using adaptive-fuzzy logic control, Aliabad Katoul branch. Islamic Azad University, Aliabad Katoul, Iran
8. Tahour A, Abid H, Aissaoui AG (2007) Adaptive neuro-fuzzy controller of switched reluctance motor. Serb J Electr Eng Serb J Electric Eng IEEE Conf
9. Kannan K, Padmanabhan S (2013) PI-CCC based switched reluctance generator applications for wind power generation using MATLAB/SIMULINK. J Electr Eng Technol 8(2):230–237. ISSN: 1975-0102
10. Jayapragash R, Chellamuthu C (2013) GRID based switched reluctance generator applications for wind power generation on MPPT algorithm using MATLAB/SIMULINK. Eur J Sci Res 100(4):579–591. ISSN: 1450-216X
11. Suthapadmanabhan S, Kannan K (2014) High performance MPPT based on variable speed generator driven by wind power generation in battery applications. J Electr Eng Technol 9(1):205–213. (ISSN: 1975-0102)

Ensuring Data Integrity in Mobile Crowdsensing Environment Using Fuzzy Logic

Ramesh K. Sahoo, Sateesh Kumar Pradhan, and Srinivas Sethi

Abstract In the modern world, industries rely on the feedback/reviews of the users for estimating their future plan for better customer care services and customer relationship management. The evaluations and follow up achievement can be computed as +ve/−ve types of feedback or review. This work attempts to present a model using fuzzy logic over mathematical model that will outperform the categorisation of customers or users feedback using ratings given by users or customers to ensure data integrity in mobile crowdsensing environment. In this work, customers or users can provide feedback or review for the location using web-based applications or Android Application, that will be stored in a cloud environment. This data-set will be analysed using fuzzy logic to isolate genuine reviews to maintain data integrity which may be used for different types of real-time applications such as tourism, medical, educations, among a few other applications and also categorise the customers or users as honest, malicious and suspicious.

1 Introduction

Data integrity reflects consistency, accuracy and safety of data in database. Data collection and storage in a proper way to ensure accuracy and reliability is a part of physical data integrity whereas logical data integrity verifies accuracy of data in the specific context. It ensures that whenever data will be analysed it will provide same information. It is required to remove invalid and fake data that can compromise the information and also it is necessary to identify users who want to use it for malicious purpose to change the information. It can be used for different real-time applications

---

R. K. Sahoo (✉) · S. K. Pradhan
CS, Utkal University, Bhubaneswar, India
e-mail: ramesh0986@gmail.com

S. Sethi
CSEA, IGIT Sarang, Dhenkanal, India

223

where analysis of vast data-set is required. Fake review analysis can be considered as one of the study in this kind of application. Feedback or reviews given by users after using a product or experiencing with services are considered for certain information about product or services that should be accurate for further use. This information has been obtained by analysing the reviews or feedback delivered by various users, and as per purpose or intention, a user can give a fake or genuine review. In current era, users blindly trust the reviews given by other users. So it must be genuine and reliable, but due to some users who give fake reviews make the reviews unreliable. Hence, it has been required to analyse feedback to detect fake or correct reviews and isolate fake reviews as well as the users who delivered fake review. Only genuine reviews can give a reliable information. This will ensure the data integrity in the mobile crowdsensing paradigm, whereas mobile crowdsensing [1] is a process of data collection, integration and analysis of large and heterogeneous data obtained from variety of sources in urban spaces, such as smartphone, laptop, desktop, tablet, mobile devices, sensors, buildings, vehicles and human.

Information that has been obtained by proper analysis of feedback or reviews received from valid and honest users or customers only may be taken as actual, reliable and trustable. Isolation and identification of actual reviews from real-time data-set is one of the challenging jobs in the current scenario. Various techniques such as: sentiment analysis, convolutional neural network (CNN), classification, deep learning and machine learning can be employed for text processing to detect and isolate actual reviews from real-time data-set. Not only reviews or feedback but also activity of users or customers are also need to be analysed to identify actual reviews/feedback and Honest users.

In this work, the model has been concentrated on detection and isolation of actual reviews to ensure data integrity by removing invalid and fake reviews so that information will be accurate and reliable. It also categorised the users as honest, malicious or suspicious from the collected data. Information obtained by analysis of feedback or reviews received from honest and legitimate user only will be provided to people who use the products or services. It will help to ensure data integrity in mobile crowdsensing environment by considering genuine reviews came from honest user only to provide accurate and reliable information. In this model, the review has been obtained from various users through web-based application and android app using various electronic gadgets such as smartphone, tablet and laptop in mobile crowdsensing environment exclusively for this purpose. Then obtained review has been stored in the cloud platform and analysed using Fuzzy Logic concept through MATLAB (Simulation Software) for categorisation of users or customers and identification of honest feedback/reviews as well as honest registered and participated users.

The remaining section of the paper has been discussed with background in Sect. 2, followed by methodology in Sect. 3. Analyses on experimental result have been done in Sect. 4. The paper has been concluded in Sect. 5.

2   Background

In this section, background of the paper has been discussed using various related research articles.

Fake feature framework has been discussed in [2] for the detection of fake feedback/reviews to extract and characterise feature of review based on registered and participated users and feedback/reviews. Author highlight here, by only analysing review (in text format) will not work so personal, social, review activity and trustworthiness of user is also required for analysis of reviews. For the detection of outlier review, authors in [3] discussed a model that is based on comments and reviews given for the product by various users using data obtained from china, amazon data-set. Instead of analysing the current review only it also analyse the past review of product received from different sources to identify fake reviews. Cognitive, affective, perceptual and social linguistic characteristics of the psychological process of reviewer has been discussed in [4] to establish a bond with fake reviews by estimating the influence of distance, time and location of reviewers on the reviews.

Machine learning, fuzzy logic and artificial intelligence has been applied to large social media-based data-set to detect spam through neural network multilayer perception and to overcome limitations of supervised learning algorithm using unsupervised approach. Fuzzy logic can handle large data-set in less time to reduce cost, time and need of complex software to detect spammer [5]. For opinion spam detection, fuzzy modelling-based solution has been proposed which contains 81 fuzzy rules generated using FSL detection algorithm, 4 linguistic variable as input for fuzzy set and fuzzy ranking evaluation algorithm to detect suspicious group and it achieves 80.77% of accuracy [6]. For spam detection and email classification, Interval type 2 fuzzy set can be considered as an effective technique that powered the users to have more control over different types of spam and also allow to personalise it [7].

Authors in [8] discussed a mathematical model for management of trust or reliability in secure spectrum sensing in CRN. Mathematical model for detection of fake reviews as well as categorisation of user as Honest, suspicious and Malicious has been developed in [9].

3   Proposed Methodology

3.1   Concept

An smart phone Application and web-based applications have been developed that can be used as a digital guide for different applications, which works through any electronics gadgets in mobile crowdsensing environment to maintain data integrity. The proposed system has following objectives.

1. Obtaining review data from various users about their experience.
2. Storing and analysing data in the cloud platform to compute the reputation level of a place based on review and rating.
3. Categorisation of genuine and fake data by computing user's trustworthiness. Valid, accurate and actual data may be taken for developing the proposed model. It helps to ensure integrity of data by removing fake or invalid or inaccurate data from data-set so that information will be generated from genuine reviews only and this information will accurate, complete and reliable.

3.2 Architecture

In the proposed model, initially any user will download and install the application in smartphone or use web-based application to registered himself by providing information such as name, cell phone number, gender, email id, current address, date of birth and profession in the mobile crowdsensing environment. Review Data will be obtained from different registered users about different location through the mobile devices such as smartphone, tablet and laptop in both off-site and on-site mode and it will be placed in cloud environment. Review data-set will be analysed using fuzzy logic and proposed mathematical model to detect and isolate fake reviews, and also categorised users as malicious, suspicious and honest using various performance metrics like reliability, incentive, honesty level. Any registered/unregistered user can get required information on any location through the app or web-based application. The information about location will be delivered to forthcoming users by analysing reviews received from honest users only. Therefore information will be accurate and reliable and it helps to ensure integrity of data in mobile crowdsensing environment.

3.3 Framework for Collection of Data

Data of user will be received from various registered and participated users as feedback or review on the experiences in on-site or off-site mode. In on-site mode, the registered and participated user will deliver feedback/review on the current location of his own. In off-site mode, the registered and participated user will deliver feedback/review on any recently travelled location from his comfort point. information such as unique ID of user, address, coordinates of location in longitude and latitude, any message regarding that location, communicative language and rating in the range of 1–5 for that location has been obtained as review. It also contains other features such as: dislike, like, rectification required and average which is received as information about the visited location through electronic gadgets such as smartphone, tablet and laptops. This information has been stored in a cloud database. Finally, it will be analysed to identify fake feedback/reviews and isolate genuine reviews to ensure data integrity in mobile crowdsensing environment and estimate users' honesty level. In

the proposed work, 1415 number of review data have been collected from 51 users for 42 locations. It has been observed that out of 1415 reviews, 585 reviews were received from honest user, 194 reviews were obtained from malicious users, and 636 reviews came from suspicious user.

3.4 Pre-processing of Data with Analysis

Collected data available in the cloud platform will be retrieved and transformed as text file in comma-separated value (CSV) format. This real-time data-set will be analysed through simulation software (MATLAB).

3.5 Computation of Average Rating of Locations

Average rating of location can be the approximate actual rating of location. It is computed using review given by all user for that location using maximum likelihood estimation technique. It can be considered as the actual rating of that place approximately. Total rating as well as estimated rating of location has been determined using equation mentioned below [9],

$$R_1 = \sum_{u=0}^{n} L_u \qquad (1)$$

$$E_1 = \frac{R_1}{n} \qquad (2)$$

In Eq. 1, user is represented by u that is in the range of 0 to n, and rating delivered by registered and participated user for the location l is represented by L_u. Total rating given by all registered and participated users for the location is represented by R_1.

In Eq. 2, Estimated rating (ERT) of location has been computed by considering the mean of R_1 and represented by E_1.

3.6 Reliability Level User

Any user will give a rating to a location and using fuzzy logic it will be compared with estimated rating (ERT) of that location obtained using Eq. 2 to compute status of review in the range of 0–1 given by user. If status of review is greater than a threshold value (δ) than it will be taken as honest review and reliability level of user will enhance otherwise fake feedback/review and reliability level of registered and participated user will reduce. If review is honest, then no. of correct review (NCR)
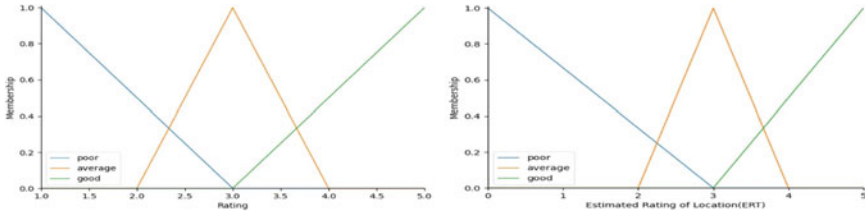
(a) Fuzzy membership value of rating given (b) Fuzzy membership value of Estimated
by user for a location   Rating of location

Fig. 1 Fuzzy membership value for input

provided by registered and participated user will be increased by 1. Else, no of fake
review will be increased by 1. Reliability level of participated user will be computed
using number of correct review and number of wrong review given by users using
fuzzy logic and proposed mathematical model. It will help to identify and isolate
genuine reviews from data-set to maintain data integrity in mobile crowdsensing
environment.

Figure 1a reflect the membership values of Rating given by user in the range of
1–5 for a location using fuzzy logic. Triangular membership function (trimf) [10]
and Rating given by user represented by U_r of fuzzy logic has been used to generate
3 membership parameter poor, average and good using Eqs. 3–5.

$$\text{Rating}['poor'] = \text{trimf}(U_r, (1, 1, 3)) \tag{3}$$

$$\text{Rating}['average'] = \text{trimf}(U_r, (2, 3, 4)) \tag{4}$$

$$\text{Rating}['good'] = \text{trimf}(U_r, (3, 5, 5)) \tag{5}$$

Figure 1b reflect the membership function of Estimated Rating of location obtained
using Eq. 1 in the range of 0–5 for a location using fuzzy logic. Poor, average and good
are the 3 membership parameter has been generated using triangular membership
function (trimf) and ERT represented by E_1 using Eqs. 6–8.

$$\text{ERT}['poor'] = \text{trimf}(E_1, (0.0, 0.0, 3.0)) \tag{6}$$

$$\text{ERT}['average'] = \text{trimf}(E_1, (2.0, 3.0, 4.0)) \tag{7}$$

$$\text{ERT}['good'] = \text{trimf}(E_1, (3.0, 5.0, 5.0)) \tag{8}$$

Figure 2 reflect the membership function of status of review in the range of 0–5
for the Rating given by user for a location using fuzzy logic. Triangular membership
function (trimf) of fuzzy logic has been used to generate 3 membership parameter
low, medium and high using Eqs. 9–11.

Fig. 2 Fuzzy membership value of review status of user (output)
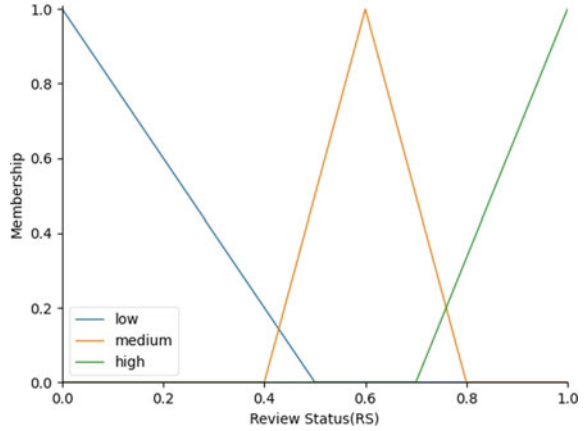
$$RS['low'] = trimf(RS, (0, 0, 0.5)) \tag{9}$$
$$RS['medium'] = trimf(RS, (0.4, 0.6, 0.8)) \tag{10}$$
$$RS['high'] = trimf(RS, (0.7, 1, 1)) \tag{11}$$

There are 9 fuzzy rules been used to compare rating given by user with Estimated Rating of location (ERT) to compute review status using fuzzy logic. Poor, average and good are the membership parameter of Rating and Estimated Rating of location whereas low, medium and high are the membership parameter of review status. Rules has been generated using membership parameter discussed above. Fuzzy rules are mentioned below.

Rule 1—if Rating is low and ERT is low then Review Status is high.
Rule 2—if Rating is average and ERT is average then Review Status is high.
Rule 3—if Rating is good and ERT is good then Review Status is high.
Rule 4—if Rating is low and ERT is good then Review Status is low.
Rule 5—if Rating is good and ERT is low then Review Status is low.
Rule 6—if Rating is low and ERT is average then Review Status is medium.
Rule 7—if Rating is average and ERT is low then Review Status is medium.
Rule 8—if Rating is average and ERT is high then Review Status is medium.
Rule 9—if Rating is high and ERT is average then Review Status is medium.

Using fuzzy logic rating delivered by the user for a place/location will be compared with ERT of that location using fuzzy logic to get review status and it may vary from 0 to 1. Various observation has been taken for review status ranges from 0 to 1 to detect its effect on detection of fake reviews and categorisation of user and also to find out a threshold δ for review status. if review status is greater than δ then review will be considered as correct review and number of correct review (NCR) provided by registered and participated user will increase by 1. Else, review/feedback will be treated as fake review and number of fake reviews (NFR) provided by registered and

participated user will increase by 1. NCR and NFR are used to compute reliability level of user using following equation.

$$R_u = \frac{NCR}{NCR + NFR} \tag{12}$$

In Eq. 12, reliability level of users computed using NWR and NCR has been represented by R_u.

3.7 Activeness Level of User

Active participation of user has been determined using activeness level of user. If participation of user has been observed in the review process then N_p will be incremented by 1 otherwise N_a will increase by 1. N_p represents the number of cases where user provide valuable review and N_a represent number of cases where user does not provide any review. The activeness level of the user will be computed using N_p and N_a and it is determined as per equation mentioned in Eqs. 13 and 14 [9];

$$U_p = \begin{cases} 1, & \text{if participation of user has been observed for different locations} \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

In Eq. 13, status of user involvement in the review activity has been represented by U_p for different locations. If participation of user has been observed in the review process, then U_p will be considered as 1 otherwise 0. If $U_p=1$, then N_p will be incremented by 1, otherwise N_a will be incremented by 1.

In Eq. 14, activeness level of user has been represented by A_u and it is determined using N_p and N_a as per following equation.

$$A_u = \frac{N_p}{N_p + N_a} \tag{14}$$

3.8 Incentive Level of User

Incentive is the additional benefits given to user as a reward for providing honest reviews and consistent participation in review process. Multiple reviews can be obtained from an user for a location in some time interval. Therefore location-wise suspicious level has been computed. Finally, the mean of suspicious level (location-wise) of registered and participated user will be computed to determine incentive level of user.

Triangular membership function (trimf) and Rating delivered by user for a certain location represented by R_{uL} of fuzzy logic has been used to generate 3 membership parameter poor, average and good using Eqs. 15–17.

$$\text{Rating}['\text{poor}'] = \text{trimf}(R_{uL}, (1, 1, 3)) \tag{15}$$

$$\text{Rating}['\text{average}'] = \text{trimf}(R_{uL}, (2, 3, 4)) \tag{16}$$

$$\text{Rating}['\text{good}'] = \text{trimf}(R_{uL}, (3, 5, 5)) \tag{17}$$

Rating given a user for a location R_{uL} has been compared with ERT represented by E_l using fuzzy logic to determine review status of user. Threshold of review status for honest review is δ. if review status is greater than threshold then review will be considered as correct review and number of correct review for that location (NCR$_l$) will increase by 1 otherwise review will be treated as fake and number of fake review for that location (NFR$_l$) will increase by 1. Suspicious level of user represented by S_{uL} for that location will be determined using NCR$_l$ and NFR$_l$ using Eq. 18 [9].

$$S_{uL} = \frac{\text{NFR}_l}{\text{NCR}_l + \text{NFR}_l} \tag{18}$$

Suspicious level of registered and participated User will be computed by taking mean of place/location-wise suspicious level (S_{uL}) of all locations of user using Eq. 19.

$$S_u = \frac{\sum_{L=0}^{n} S_{uL}}{n} \tag{19}$$

in Eq. 19, S_u represent Suspicious level of User and n is the number of location for which a user provide review. Finally incentive level of user represented by I_u will be determined using Suspicious level (S_u) of user in Eq. 20 [9].

$$I_u = 1 - S_u \tag{20}$$

3.9 Determination of Honesty Level of User

Weight values have been assigned to user as per his performance estimated using reliability level, suspicious level and incentive level of user as listed in Table 1 [9].

The honesty level (H_u) of user will be computed using various performance metrics such as activeness (A_u), reliability (R_u) and incentives (I_u) levels and their weight values as per mentioned in Eq. 21 [9].

$$H_u = W_1 * R_u + W_2 * A_u + W_3 * I_u \tag{21}$$

Table 1 Fixed values for weight factors

Component	Weight variable	Value
Reliability	W_1	0.5
Suspicious	W_2	0.2
Incentive	W_3	0.3

Table 2 Type of user

Honesty level (H_u)	User type
>0.7	Honest
0.4–0.7	Suspicious
<0.4	Malicious

3.10 Categorisation of User

Finally user will be categorised as Honest, malicious and Suspicious User using Honesty level (H_u) of user as per Table 2 [9].

4 Result and Discussion

In this section of the paper, the results and analysis have been discussed.

Figure 3 provides a visual comparison between Sahoo-2021 model [9] and Proposed Model in which x axis represents number of users in percentage and y axis represents type of user. It is observed that after applying fuzzy logic number of honest users has been decreased as well as number of suspicious and malicious user has been increased as compared to Sahoo-2021 Model [9].
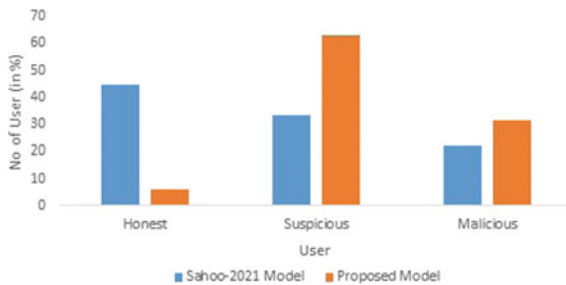
Fig. 3 Comparison between Sahoo-2021 model [9] and proposed model
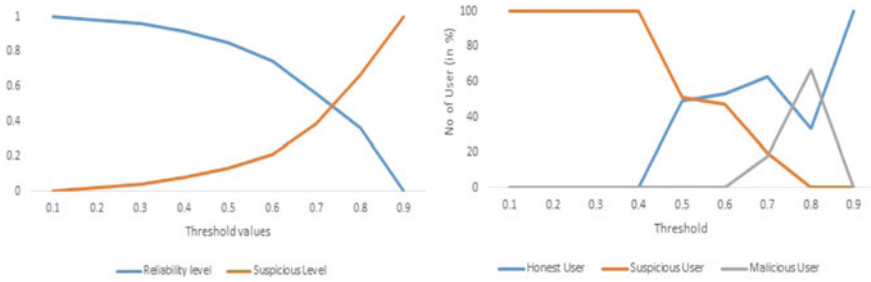
(a) Effect of review status on Reliability and Suspicious level

(b) Effect of review status on categorisation of user

Fig. 4 Threshold for review status
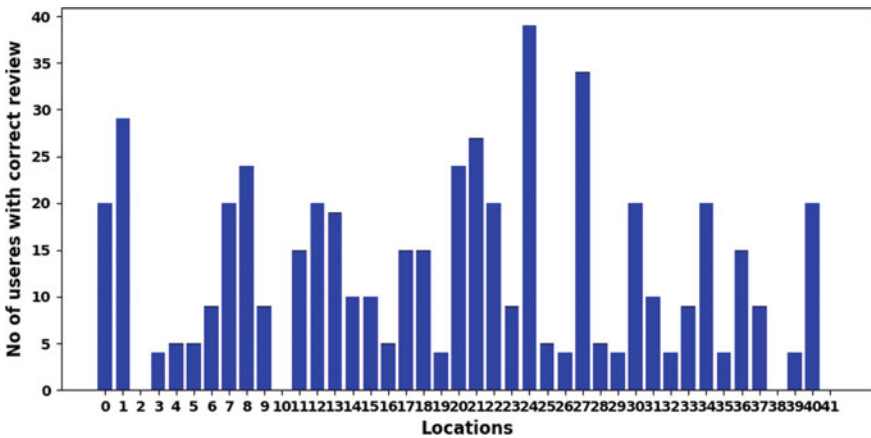
Fig. 5 No of users with correct review versus locations

Figure 4 reflect effect of review status on reliability and suspicious level of user and also in categorisation of user. review status is the output of fuzzy model which varies from 0 to 1. It is observed Figure 4a, b that for review status greater than 0.4 reliability level of user decreases and suspicious level increases and both of them intersect and review status 0.75. For review status 0–0.4 all users are considered as suspicious user but number of honest user increases after value of review status greater than 0.4 and number of malicious users also increase after review status greater than 0.6 and it is minimised to 0 after review status greater than 0.8. all three types of users are available at review status 0.7. Therefore it is concluded that threshold for review status (δ) should be 0.7.

In Figs. 5 and 6, number of users who have given correct review as well as fake review for various location has been reflected. it has been observed that for some
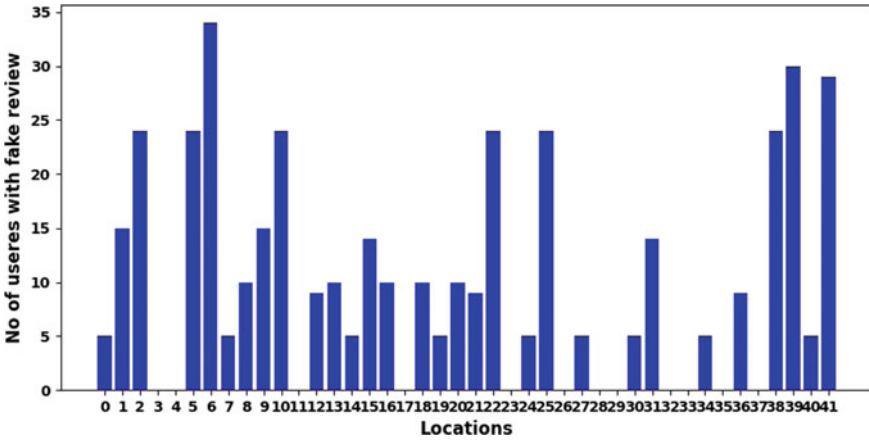
Fig. 6 No of users with fake review versus locations
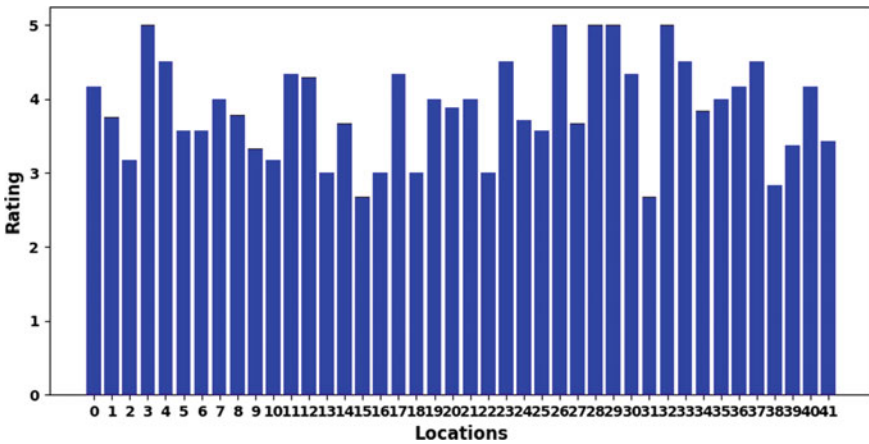
Fig. 7 Estimated rating of locations

locations like location id 3, 23, 33 etc., number of fake reviews are more than correct review also it has been observed for each and every location some reviews are fake as well as correct.

In Fig. 7, Rating axis represent estimated rating of location (ERT) in between 0 and 5 whereas unique locations numerically assigned from 0 to 41 has been represented by locations axes. The estimated rating of location (ERT) may be assumed as the real rating of the place/location approximately that is computed from review data obtained from visitors of particular location.
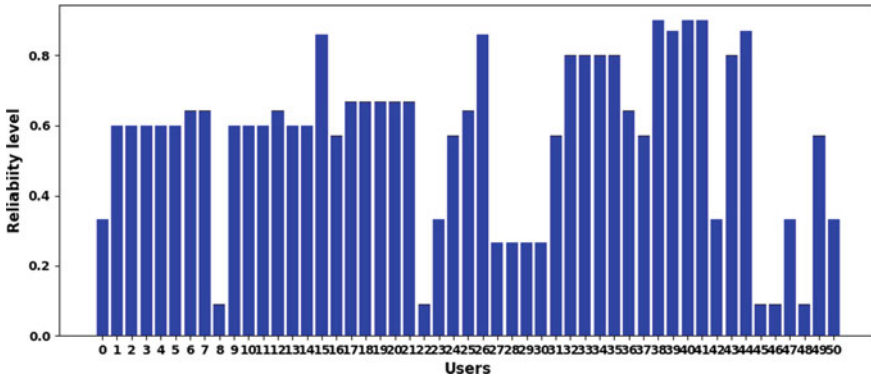
Fig. 8 Reliability level of users
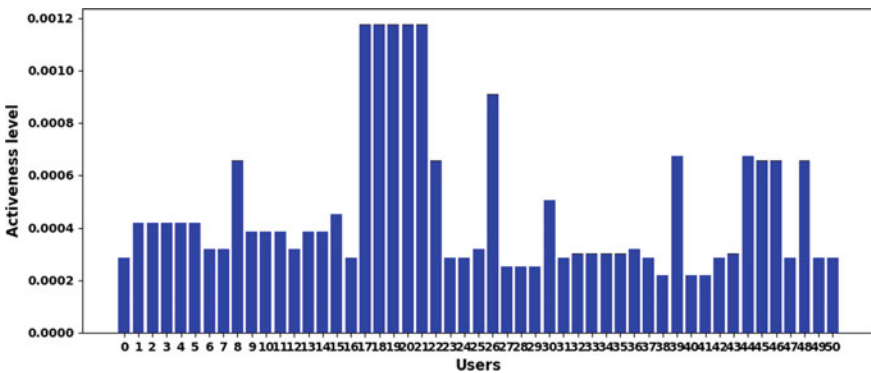
Fig. 9 Activeness level of users

Figure 8 demonstrate reliability level of users having unique id ranges from 0 to 50 computed as per their given review. It reflects how much extent an user can be trustworthy as per his/her review. It is increased when review given by the user is correct and decreased when the review given by user is fake. Therefore, due to correct reviews reliability level of user will increase whereas it will decrease when given review is fake.

Figure 9 represents activeness level of various users having unique id ranges from user id 0 to 50. It reflects continuous involvement of users in the review process and it is in the range of 0–1.

Figure 10 reflects incentive level of various users having unique user id ranges from 0 to 50. It demonstrates the amount of incentive provided to users in the range of 0–1 as per user's reliability level and activeness level in the feedback/review mechanism. Consistent and reliable user will be rewarded with higher incentives. Incentive level is positively influenced by the no. of honest reviews delivered by registered and participated users and negatively influenced by the no. of fake feedback/reviews.
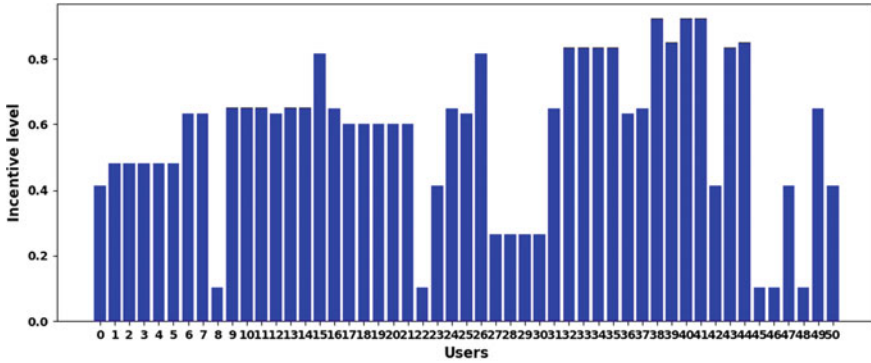
Fig. 10 Incentive level of users
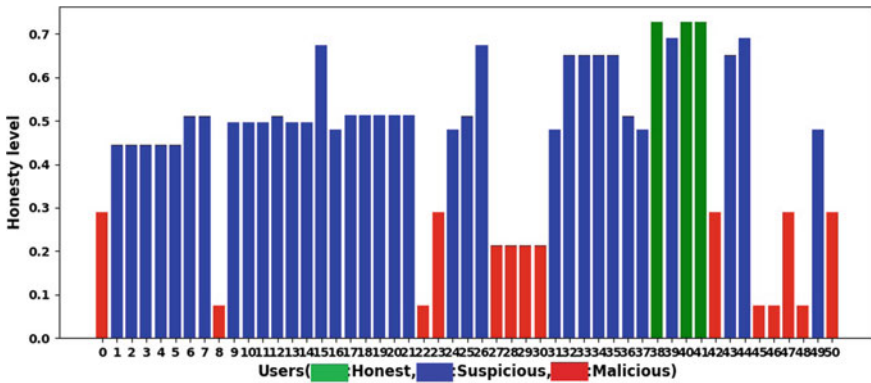
Fig. 11 Honesty level of users

Figure 11 demonstrate Honesty level of different users in the range of 0–1 and fixed numerical range from 0 to 50 is used to assign unique id to various users. Based on reliability, activeness and incentive level of user and constant weight factor, Honesty level of user has been determined. It categorises various users into three categories: honest user represented by green colour, whereas suspicious user represented by yellow colour, and finally malicious user represented by red colour.

5 Conclusion

In this work, a novel idea is placed to ensure data integrity in mobile crowdsensing environment by categorising the reviews such as: actual and fake using fuzzy logic through different member functions. Further, categorisation of the reviewers as honest, malicious and suspicious is also placed using mathematical model. In this

context some of the users are replaced as malicious and suspicious from the honest users which is placed in Sahoo-2021 model [9]. The ERT of various locations is computed by analysing the data-set that contains feedback uploaded by various users. Estimation of Location may be considered as the actual rating of the location approximately. The user reliability is used to describe the trustworthiness of reviews and the users. In this context the review and the users may be reliable which ensure Data Integrity. The activeness level of users represents consistency and reliability of users in review observation. Incentives/additional benefits will be provided to various users as per their activeness and reliability level which is reflected in Figs. 9 and 10. Finally honest user has been estimated using reliability, activeness level and mentioned incentives to users as per their performance of user to reflect trustworthiness of user. Reviews or feedback delivered by different users are considered to be analysed to identify and isolate the real reviews which helps for elimination of fake review and isolation genuine reviews to maintain data integrity in mobile crowdsensing environment.

References

1. Capponi A, Fiandrino C, Kantarci B, Foschini L, Kliazovich D, Bouvry P (2019) A survey on mobile crowdsensing systems: challenges, solutions, and opportunities. IEEE Commun Surv Tutor 21(3):2419–2465
2. Barbado R, Araque O, Iglesias CA (2019) A framework for fake review detection in online consumer electronics retailers. Inf Process Manag 56(4):1234–1244
3. Liu W, He J, Han S, Cai F, Yang Z, Zhu N (2019) A method for the detection of fake reviews based on temporal features of reviews and comments. IEEE Eng Manag Rev 47(4):67–79
4. Li L, Yee LK, Lee M, Yang S-B (2020) Unveiling the cloak of deviance: linguistic cues for psychological processes in fake online reviews. Int J Hosp Manag 87:102468
5. Tehlan P, Madaan R, Bhatia KK (2019) A spam detection mechanism in social media using soft computing. In: 2019 6th international conference on computing for sustainable global development (INDIACom). IEEE, pp 950–955
6. Dhingra K, Yadav SK (2019) Spam analysis of big reviews dataset using fuzzy ranking evaluation algorithm and Hadoop. Int J Mach Learn Cybern 10(8):2143–2162
7. Ariaeinejad R, Sadeghian A (2011) Spam detection system: a new approach based on interval type-2 fuzzy sets. In: 2011 24th Canadian conference on electrical and computer engineering (CCECE). IEEE, pp 000379–000384
8. Kar S, Sethi S, Sahoo RK (2017) A multi-factor trust management scheme for secure spectrum sensing in cognitive radio networks. Wireless Pers Commun 97(2):2523–2540
9. Sahoo RK, Sethi S, Udgata SK (2021) A smartphone app based model for classification of users and reviews (a case study for tourism application). In: Intelligent systems. Springer, pp 337–348
10. Matlab (2021) Triangular membership function. https://www.mathworks.com/help/fuzzy/trimf.html. Accessed 21 Sept 2021

Relativistic Mean-Field Treatment to the Fusion Cross-Section of the Reaction ^{12}C + ^{20}Ne Using Hill–Wheeler and Morse Transmission Coefficients

Shilpa Rana ⓘ **, Joshua T. Majekodunmi** ⓘ **, Nishu Jain** ⓘ **, M. Bhuyan** ⓘ **, Raj Kumar** ⓘ **, and Suresh Kumar Patra** ⓘ

Abstract The transmission through the Coulomb barrier is essential to determine the cross-section of various fusion reactions. In this analysis, we have compared the cross-section of ^{12}C + ^{20}Ne reaction calculated using the analytical expressions for transmission coefficients through the parabolic and Morse barrier. The nuclear potential for both approaches is generated using the R3Y nucleon–nucleon (NN) potential and the relativistic mean-field (RMF) densities for NL3* parameter set within the double folding approach. The fusion cross-section is obtained from the $\ell-$summed Wong model. It is observed that at the below-barrier energies, the Morse transmission coefficient gives a better overlap with the experimental data. At the above-barrier energy region, the cross-section obtained from both the parabolic and Morse barrier approximations almost overlaps with each other and fits the experimental data nicely.

S. Rana (✉) · N. Jain · R. Kumar
School of Physics and Materials Science, Thapar Institute of Engineering and Technology, Patiala 147004, India
e-mail: srana60_phd19@thapar.edu

J. T. Majekodunmi
Faculty of Applied and Human Sciences, Institute of Engineering Mathematics, Universiti Malaysia Perlis, Arau 02600, Perlis, Malaysia

M. Bhuyan
Department of Physics, Faculty of Science, Center for Theoretical and Computational Physics, University of Malaya, Kuala Lumpur 50603, Malaysia

Institute of Research and Development, Duy Tan University, Da Nang 550000, Vietnam

S. K. Patra
Institute of Physics, Sachivalaya Marg, Sainik School, Bhubaneswar 751005, India

Homi Bhaba National Institute, Anushakti Nagar, Mumbai 400085, India

239

1 Introduction

The heavy-ion fusion reactions have fascinated nuclear physicists as these reactions are significant in exploring the nuclear structure, nuclear forces and in extending the nuclear chart [1–3]. Moreover, the light heavy-ion fusion reactions at far below-barrier energies are essential to probe the stellar nucleosynthesis process at different phases of stellar evolution [1–3]. The description of the complex fusion mechanism is given through the interaction potential containing the Coulomb, centrifugal and nuclear interaction terms. The evaluation of transmission probability through the barrier formed due to this interaction potential is mandatory to obtain the cross-section of the heavy-ion reactions [1–3].

The analytical expressions for the barrier transmission coefficient are preferred to avoid the complexities associated with its numerical evaluation. Several analytical expressions for the transmission probability obtained using different approximations for the barrier shapes are available in the literature [1, 4–7]. One of the well-known approaches is Hill–Wheeler (HW) formula [5, 6], in which the fusion barrier is assumed to be purely parabolic. Due to its simple expression for transmission coefficient, the parabolic barrier approximation has been widely adopted in the description of various fusion reactions [1, 3, 8–12]. However, the Hill–Wheeler approximation of symmetric parabolic barrier becomes inadequate at the deep sub-barrier energies due to steep fall of the parabola at large distances [1, 4]. This problem can be addressed by approximating the fusion barrier by asymmetric Morse function [7]. The expressions for the transmission coefficient through both parabolic and Morse barriers involve the barrier height, position and frequency [5–7]. To determine these quantities, an appropriate description of the nuclear interaction part of the total potential formed between the fusing nuclei is necessary. In the present analysis, this nuclear potential is estimated using double folding approach [13] furnished with densities and effective nucleon–nucleon (NN) potential [10, 14–16] calculated from relativistic mean-field (RMF) formalism [10–12, 17–21].

The RMF formalism has gained successful applications in describing the characteristics of the nuclear matter, and structural properties as well as reaction dynamics of finite nuclei [10–12, 17–21]. Thus, it will be interesting to compare the Hill–Wheeler and Morse barrier transmission coefficients given in terms of barrier characteristics obtained from RMF formalism. These transmission coefficients are further employed to evaluate the fusion cross-section for $^{12}C + {}^{20}Ne$ reaction from the $\ell-$summed Wong model [8, 9]. The calculated cross-section is also compared to the experimental data taken from [22].

2 Theoretical Formalism

As discussed above, the total potential formed between the colliding nuclei provides the basis to study their fusion mechanism. The interaction potential between the

fusing projectile and target nuclei at separation distance R with atomic numbers Z_p and Z_t, respectively, can be written as

$$V_T^\ell(R) = \frac{Z_p Z_t e^2}{R} + \frac{\hbar^2 \ell(\ell+1)}{2\mu R^2} + V_n(R). \tag{1}$$

Here, μ and $V_n(R)$ denote the reduced mass and the nuclear potential, respectively. This nuclear potential is determined employing the double folding integral [13] as,

$$V_n(\mathbf{R}) = \int \rho_p(\mathbf{r}_p)\rho_t(\mathbf{r}_t)V_{eff}\left(|\mathbf{r}_p - \mathbf{r}_t + \mathbf{R}| \equiv r\right) d^3 r_p d^3 r_t. \tag{2}$$

Here, ρ_t and ρ_p denote target and projectile densities, respectively. The term V_{eff} symbolizes the NN interaction potential. These densities and NN potential are obtained from the well-established RMF formalism. In RMF formalism, the nucleons are assumed to be point-like particles interacting through the exchange of photons and mesons. The details of the phenomenological Lagrangian density and RMF equations can be found in [10–12, 17–21] and references therein. The R3Y NN potential can be obtained solving the RMF equations for mesons [10, 14–16] and is given as,

$$V_{eff}^{R3Y}(r) = \frac{g_\omega^2}{4\pi}\frac{e^{-m_\omega r}}{r} + \frac{g_\rho^2}{4\pi}\frac{e^{-m_\rho r}}{r} - \frac{g_\sigma^2}{4\pi}\frac{e^{-m_\sigma r}}{r} + \frac{g_2^2}{4\pi}re^{-2m_\sigma r}$$
$$+ \frac{g_3^2}{4\pi}\frac{e^{-3m_\sigma r}}{r} + J_{00}(E)\delta(r). \tag{3}$$

The $g_\sigma, g_\omega, g_\rho$ denote the coupling constants of the σ, ω, ρ mesons having masses $m_\sigma, m_\omega, m_\rho$, respectively. g_2 and g_3 correspond to the self-interaction of σ-mesons. These RMF parameters are obtained using the NL3* [21] parameter set. More details can be found in [10, 14–16]. The term $J_{00}(E)\delta(r)$ denotes the long-range one pion exchange potential. The RMF nuclear densities and R3Y NN potential are employed to obtain the nuclear potential using Eq. 2.

The next step is to determine the fusion probability. For this, two approximations are used for assimilating the shape of the fusion barrier. The first one is the parabolic barrier [5, 6] given by

$$V_T^{HW}(R) = V_B^\ell - \frac{1}{2}\mu\omega^2(R - R_B^\ell)^2. \tag{4}$$

Here, V_B^ℓ and R_B^ℓ are the height and position of the fusion barrier (see Eq. 1) and are determined using the following conditions:

$$\left.\frac{dV_T^\ell}{dR}\right|_{R=R_B^\ell} = 0. \tag{5}$$

$$\frac{d^2 V_T^\ell}{dR^2}\bigg|_{R=R_B^\ell} \le 0. \tag{6}$$

The curvature of fusion barrier ($\hbar\omega_\ell$) is calculated at $R = R_B^\ell$ as,

$$\hbar\omega_\ell = \hbar[|d^2 V_T^\ell(R)/dR^2|_{R=R_B^\ell}/\mu]^{\frac{1}{2}}. \tag{7}$$

The penetration probability for this parabolic barrier is given by Hill–Wheeler [5, 6] formula, which takes the form,

$$P_\ell^{HW} = \left[1 + \exp\left(\frac{2\pi(V_B^\ell - E_{c.m.})}{\hbar\omega_\ell}\right)\right]^{-1}. \tag{8}$$

The Hill–Wheeler approximation provides a simple and numerically inexpensive expression for the transmission probability. However, this approximation of parabolic fusion barrier fails at deep sub-barrier energies for fusion reactions involving lighter nuclei [1, 4]. As an alternative, we have also considered the Morse function [7] approximation for the fusion barrier. Morse function gives an asymmetric fusion barrier and is expressed as,

$$V_T^{Morse}(R) = V_B^\ell[2e^{-(R-R_B^\ell)/\alpha_M} - e^{-2(R-R_B^\ell)/\alpha_M}], \tag{9}$$

α_M is the Morse parameter given by,

$$\alpha_M = [2V_B^\ell/|d^2 V_T^\ell(R)/dR^2|_{R=R_B^\ell}]^{\frac{1}{2}}. \tag{10}$$

The analytic expression for transmission coefficient through the Morse barrier is written as [7],

$$P_\ell^{Morse} = \frac{1 - \exp(-4\pi\alpha)}{1 + \exp[2\pi(\beta - \alpha)]}. \tag{11}$$

Here, $\alpha = k\alpha_M$ and $\beta = \frac{\sqrt{2\mu V_B^\ell}}{\hbar}\alpha_M$. $E_{c.m.}$ is energy of projectile target system in center of mass frame and $k = \frac{\sqrt{2\mu E_{c.m.}}}{\hbar}$.

Finally, the cross-section is calculated from the extended ℓ−summed Wong model [9], which accounts for the actual modifications entering in the fusion barrier due to its dependence upon angular momentum. The fusion cross-section is given by

$$\sigma(E_{c.m.}) = \frac{\pi}{k^2} \sum_{\ell=0}^{\ell_{max}} (2\ell + 1) P_\ell(E_{c.m.}). \tag{12}$$

The ℓ_{max} values are extracted at $E_{c.m.} > V_B$ from sharp cutoff model [23].

3 Calculations and Results

The cross-section for ^{12}C + ^{20}Ne reaction, obtained using the Hill–Wheeler [5, 6] and Morse approximations [7] for the transmission coefficient is examined in this section. To determine the parameters for both the parabolic (see Eq. 4) and the Morse (see Eq. 9) curves, we first need the nuclear potential. This nuclear potential is estimated by folding the RMF densities over the relativistic R3Y NN potential. The densities for fusing nuclei and the effective NN potential are calculated using the self-consistent RMF formalism for the NL3* parameter set. The R3Y NN potential given in Eq. 3 is comparable to the non-relativistic M3Y NN potential [10, 14–16]. In our previous studies [10, 11], we compared the cross-section obtained from M3Y and R3Y NN potentials and observed that the R3Y potential provides better overlap with the experimental data. Since the main motive of the this analysis is to compare the results of Hill–Wheeler and Morse transmission coefficients, we have only considered the R3Y NN potential here.

Figure 1 shows the radial distributions of RMF (NL3*) neutron ρ_N (solid lines) and proton ρ_N (dashed lines) densities for the ^{12}C (black lines) and ^{20}Ne (blue lines) nuclei. The total density distribution is obtained by adding these neutron and proton densities (i.e., $\rho_T = \rho_N + \rho_P$) and are further employed to determine the nuclear potential within Eq. 2. Since nuclear fusion is a surface phenomenon, so the tail region of total density plays the most crucial role in describing the fusion characteristics [24].

Next, we have calculated the total interaction potential using Eq. 1. As discussed above, we have used the parabolic (Eq. 4) and Morse curve (Eq. 9) approximations for the fusion barrier. The parameters of these parabolic and Morse barriers are extracted from the actual interaction potential given by Eq. 1. Figure 2 shows the barrier region of the exact (solid black line) interaction potential as well as those generated using parabolic (dashed blue line) and Morse functions (dash-dotted red

Fig. 1 The proton (ρ_P) and neutron (ρ_N) densities obtained using RMF formalism with NL3* parameter set for ^{12}C (black lines) and ^{20}Ne (blue lines) nuclei
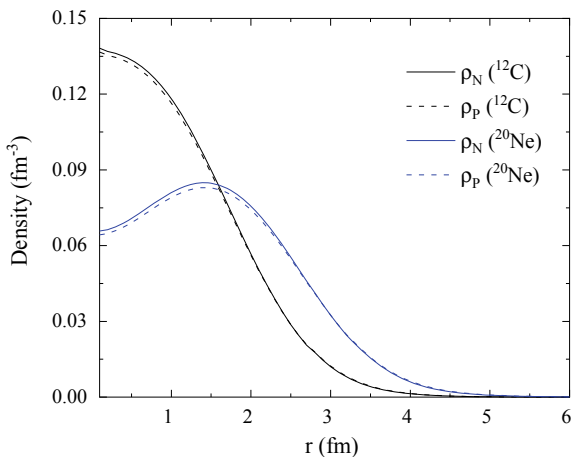
Fig. 2 Fits to the exact fusion barrier at $\ell = 0\hbar$ (solid black line) by pure parabolic (dashed blue line) and Morse curve (dash-dotted red line). See text for details
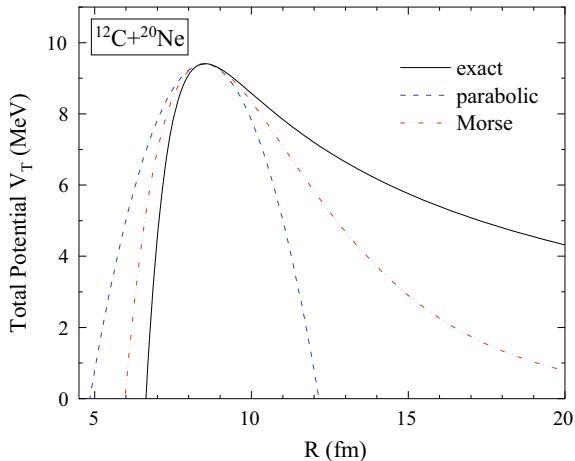
line). It is worth mentioning that all the three fusion barriers are here plotted for $\ell = 0\hbar$ (s-wave). It can be noted From Fig. 2, that, unlike the actual fusion barrier, the parabolic barrier is symmetric around the barrier position. Moreover, the structure of the barrier approximated by the parabolic function differs from the actual barrier at the large separation distance (R). These variations from the actual barrier are less for the case of Morse function, which generates an asymmetric fusion barrier that falls less steeply at larger R [7]. It is also observed that at energies near the fusion barrier, any of the approximations considered for calculating the transmission coefficient will yield similar results.

The penetration probability through a fusion barrier depends upon its characteristics. The penetrability for the parabolic and Morse fusion barriers are obtained using the expressions in Eqs. 8 and 11, respectively. These transmission coefficients are further used to obtain the cross-section for $^{12}C + ^{20}Ne$ reaction from $\ell-$summed Wong model.

Figure 3 shows the cross-section σ (mb) $E_{c.m.}$ (MeV) obtained using the Hill–Wheeler (black Solid line) and Morse barrier (red dashed line) transmission coefficients. The experimental cross-section [22] is also given for the comparison. It is noticed that at the above-barrier energy region, the cross-section given by both approximations almost overlap and provides a satisfactory fit to the experimental data. At the below-barrier energy region, the Morse transmission coefficient gives a better overlap with the experimental data in comparison to that given by the Hill–Wheeler approximation. Further, it is also observed that the difference between the cross-section given by parabolic and Morse approximate fusion barriers increase as we move toward the $E_{c.m.} << V_B$. This is because the variations of approximate barriers from the actual barrier shape increase with the increase in the separation distance R (see Fig. 2), and fusion cross-section for light systems at much below-barrier energies is sensitive to the barrier characteristics at larger R [1, 4]. This infers that the cross-section given by the analytical expression of transmission coefficients through

Fig. 3 Fusion cross-section σ (mb) calculated within $\ell-$summed Wong model using the Hill–Wheeler (solid black line) and Morse barrier (dashed red line) transmission coefficients. The experimental data (black circles) is extracted from [22]
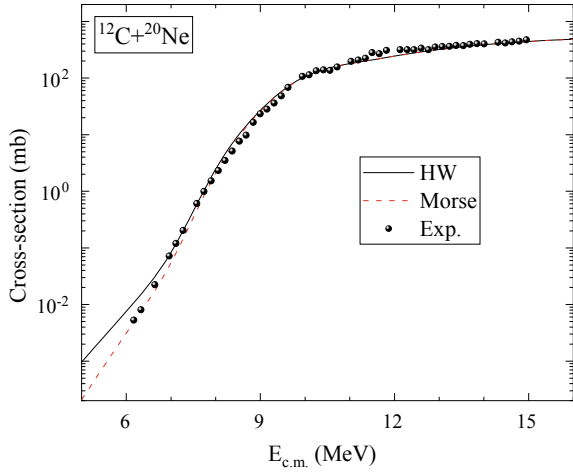
approximate fusion barrier will differ from the experimental data at energies lying far below the fusion barrier. For such cases, the semi-classical WKB approach [1, 4] which uses the exact fusion barrier, becomes more reliable. This method is not used in the present analysis as the experimental cross-section for $^{12}C + ^{20}Ne$ is not available at far sub-barrier energies. A more systematic study involving other light heavy-ion fusion reactions will be carried soon.

4 Summary and Conclusions

The fusion cross-section for $^{12}C + ^{20}Ne$ reaction calculated using the Hill–Wheeler and Morse transmission coefficients is compared at below and above-barrier energy regions. The nuclear interaction potential part of the fusion barrier is obtained using the RMF densities and R3Y NN potential within the double folding approach. At below-barrier energies, the Morse transmission coefficient is observed to give a better overlap with experimental data than the Hill–Wheeler transmission coefficient. However, the cross-section given by both approximations almost overlaps around and above the barrier energy regions. This indicates that the asymmetric fusion barrier given by the Morse function is a better approximation to study the fusion mechanism of fusion reactions involving light nuclei at below-barrier energies. A more comprehensive study involving more fusion reactions, especially those with astrophysical significance at far below-barrier energy regions, is under process and will be communicated shortly.

References

1. Canto LF, Guimaraes V, Lubian J, Hussein MS (2020) Eur Phys J A 56:281
2. Dasgupta M, Hinde DJ, Rowley N, Stefanini AM (1998) Annu Rev Nucl Part Sci 48:401
3. Shukla A, Patra SK (eds) (2020) Nuclear structure physics, Chap 5. CRC Press, Boca Raton
4. Toubiana AJ, Canto LF, Hussein MS (2017) Braz J Phys 47:321
5. Hill DL, Wheeler JA (1953) Phys Rev 89:1102
6. Ford KW, Hill DL, Wakano M, Wheeler JA (1959) Ann Phys 7:239
7. Ahmed Z (1991) Phys Lett A 157:1
8. Wong CY (1973) Phys Rev Lett 31:766
9. Kumar R, Bansal M, Arun S, Gupta RK (2009) Phys Rev C 80:034618
10. Bhuyan M, Kumar R (2018) Phys Rev C 98:054610
11. Bhuyan M, Kumar R, Rana S, Jain D, Patra SK, Carlson BV (2020) Phys Rev C 101:044603
12. Rana S, Kumar R, Bhuyan M (2021) Phys Rev C 104:024619
13. Satchler GR, Love WG (1979) Phys Rep 55:183
14. Singh BB, Bhuyan M, Patra SK, Gupta RK (2012) J Phys G Nucl Part Phys 39:025101
15. Sahu BB, Singh SK, Bhuyan M, Biswal SK, Patra SK (2014) Phys Rev C 84:034614
16. Lahiri C, Biswal SK, Patra SK (2016) Int J Mod Phys E 25:1650015
17. Vretenar D, Afanasjev AV, Lalazissis GA, Ring P (2005) Phys Rep 409:101
18. Meng J (2016) Relativistic density functional for nuclear structure. In: International review of nuclear physics, vol 10. World Scientific, Singapore
19. Ring P (1996) Prog Part Nucl Phys 37:193
20. Reinhard PG, Rufa M, Maruhan J, Greiner W, Friedrich J, Phys Z (1986) Z Phys A Atom Nucl 323:15
21. Lalazissis GA, Karatzikos S, Fossion R, Pena Arteaga D, Afanasjev AV, Ring P (2009) Phys Lett B 671:36
22. Hulke G, Rolfs C, Trautvetter HP (1980) Z Phys A 297:161
23. Beckerman M, Ball J, Enge H, Salomaa M, Sperduto A, Gazes S, DiRienzo A, Molitoris JD (1981) Phys Rev C 23:1581
24. Gupta RK, Singh D, Greiner W (2007) Phys Rev C 75:024603

Classification of Autism Spectrum Disorder Using Deep Learning

S. Preethi, A. Arun Prakash, R. Ramyea, S. Ramya, and D. Ishwarya

Abstract Autism spectrum disorder (ASD) is a condition that progressively develops during the growing stage of a child. The behavioural markers vary widely from difficulty in social interactions, speech, and converging to a pattern of behaviours. As there is no specific early diagnosis available for ASD with the advanced deep learning models, we propose the 2D convolutional neural network (CNN)-based model to differentiate the ASD patients from that of control patients (TC). To train and test the proposed architecture, the following data set, namely Autism Brain Imaging Data Exchange (ABIDE) is considered. The data set contains totally 1112 subjects, out of which, the proposed work takes 233 images. The data set consist of 104 ASD patients and 129 control patients. Then, each sample is split into sagittal, coronal and axial plane to train the model. The network is trained with AdaDelta optimization algorithm considering cross entropy loss function. The CNN model is tested with the all the samples from the test set. From all the planes, sagittal plane gives the highest training accuracy of 74.19% while the combination of all the planes and coronal plane exhibits almost similar accuracy in both the sets.

1 Introduction

Despite the commonality, the degree of impact of ASD and the symptoms vary to a smaller extent from patient to patient [1]. However, ASD usually exhibits symptoms

---

S. Preethi (✉) · A. Arun Prakash · R. Ramyea · S. Ramya · D. Ishwarya
Department of Electronics and Communication Engineering, Kongu Engineering College, Erode, India
e-mail: preethi.s.ece@kongu.edu

A. Arun Prakash
e-mail: arunprakash.ece@kongu.edu

R. Ramyea
e-mail: ramyea.ece@kongu.edu

in early childhood and strong signs at around 2–3 years of age. On the contrary, it is also possible for a kid with autism grow normally beyond 3–5 years and the show strong signs of ASD. It is estimated that one in every fifty nine kids have autism as reported by the Centers for Disease Control and Prevention (CDC). Moreover, it is also observed that boys are more prone to ASD than girls; autism is a lifelong condition.

The common symptoms or signs exhibited by the ASD patients are trouble in understanding other's feeling, repeating the actions again and again, and avoiding eye contact. Therefore, diagnosing or screening for potential ASD can be difficult as there are no clinically validated medical tests such as a blood test available [2]. The only way for doctors to make a conclusion is by looking for behavioural signs over a period of time and developmental stages. Sometimes, ASD can be detected at the month of 18 or younger. Due to these limitations of diagnosis of autism spectrum disorder, researches have attempted to classify the patient using some deep learning algorithms. To make these practically feasible solution, they used Autism Brain Imaging Data Exchange (ABIDE) data set. This data set comprises of magnetic resonance imaging (MRI) images. From the several proposed studies, it is mentioned that the ASD patients have a unique neural connections and hence can be used to early diagnosis the ASD [3].

In line with, in this study, we consider a convolutional neural network (CNN) architectures and cast the problem as binary classification problem. A CNN is also known as ConvNet. Unlike deep neural networks (DNN), CNN preserves the spatial connectivity of pixels in input image used for training the model. Therefore, they are able to extract the complex features to make a decision. For each features, the network learns a value for a bias and weights to different aspects or objects in an image to differentiate the image from the other. In ConvNet, the pre-processing essential for the process is very less compared to other classification algorithms. The research groups are autism and control patients. For training and testing, totally 233 images are used. The autism group contains 104 images and control group contains 129 images. The data set taken is ABIDE data set which comprises of MRI images. The image format is NIFTI. The MRI image consists of three planes such as sagittal, coronal, and axial. In this study, each plane is separated and used as a separate image. For each plane, the accuracy is predicted and also finally all the planes are used for the prediction of accuracy.

2   Literature Review

Heinsfeld et al. [3] proposed a deep learning model for classifying the autism spectrum disorder patients from the control patients. For classification, the brain imaging data set of ABIDE is used to analyse the brain patterns of the functional connectivity. To obtain the lower dimensional features of the data set, two stacked denoising autoencoders are used. The results were then obtained by the comparison of the model with the different classifiers such as classic support vector machine and k-nearest

neighbour classifier. It is observed from the results that the areas which differentiate the autism patients from the control patients are paracingulate gyrus, middle temporal gyrus and supramarginal gyrus. The patterns also show the anti-correlation between the anterior and posterior areas. This work achieved an accuracy of seventy per cent on training set and fifty per cent on test set.

Rathore et al. [1] proposed the classification method for autism spectrum disorder using topological features and deep learning. The three different topological features are analysed such as persistent diagrams, persistent images and persistent landscapes for the classification using the support vector machine, random forest and neural network. They also used the hybrid approach which combines the topological features with the functional correlation and obtain an accuracy of 69.2% even with the simple three-layer neural network.

Anirudh et al. [2] proposed the diagnosis of autism spectrum disorder using the Eigen values of brain network. In the resting state functional MRI images of the ABIDE data set, the pre-processing such as analysis of functional neuroimages and functional magnetic resonance imaging is applied. A brain network is constructed with the 264 regions from the fMRI images using the parcellation scheme. These features are then defined using 264 Eigen values and three other features are defined using network centralities. From this features, 64 discriminate features are obtained by using the feature selection algorithm. Then, these features are used to train a various model such as linear discriminant analysis, k-nearest algorithm, logistic regression, support vector machine and neural network. From this, a linear discriminant analysis obtained a highest accuracy of 77.7%.

El Gazzar et al. [4] proposed the bootstrapping version of graph convolutional neural network for the classification of autism spectrum disorder. This method reduces the sensitivity by using the ensemble of weekly trained G-CNN for the given graph construction. First, the bootstrapping method is used to graph an ensemble which establishes the connection in a large population graphs among the subjects. Then, G-CNN gives the computing framework to acquire interferences in the graph by utilizing the subject-specific image features in the form of function which is defined at the nodes of population. For classification, the model is focussed on the time series obtained from the rs-fMRI. The model outperforms the linear support vector machine and G-CNN method.

Mellema et al. [5] proposed the diagnosis of autism spectrum disorder using personal characteristics data from the ABIDE data set and optimised machine learning models. In the personal characteristics data, the predictive power is analysed using six characteristics such as age, three individual measures of intelligent quotient, sex and handedness. The data set is used to train nine different models. For training and testing the models, k-fold cross validation scheme is used. From the analysis, neural network model outperforms the other models with a mean AUC (standard deviation) of 0.646 (0.005) followed by k-nearest neighbour of 0.641 (0.004).

Sherkatghanad et al. [6] proposed the simple 1D convolutional neural network for the classification of autism spectrum disorder using the ABIDE data set. The rs-fMRI images from the data set are converted into the time series for progressing using 1D CNN. The full-time courses are maintained as a source of features and then sum it

up as an ensemble of voxels. The voxels are then grouped by several anatomical and functionally derived atlases such as automated anatomical labelling, Schaefer-100, Schaefer-400 and Harvard–Oxford probabilities. The matrices are used as an input to the 1D CNN with each channel corresponds to the different region of interest. Then, the binary classification is performed between the control and autism patients, and the error obtained is back propagated. From the various atlases used, the Harvard–Oxford probabilities outperform with an accuracy of 65.1% followed by Schaefer-400 of 64.3%.

Huang et al. [7] proposed the classification of autism spectrum disorder using multiple deep learning algorithms using the structural and functional MRI images from the IMPAC data set. The performance of various classifiers is compared to obtain the best results. From 3 categories, a total of 12 classifiers are compared such as 6 non-linear shallow machine learning models, 3 linear shallow models and 3 deep learning models. These models are optimised by performing random search on the model's hyperparameter space. From the analysis, the linear model performed better than other methods. The dense feedforward network obtains highest AUC of 0.80.

Ke et al. [8] proposed the classification of autism spectrum disorder using convolutional neural network. The CNN model is applied to the resting state fMRI from the ABIDE I data set. The data used is extracted from the functional connectomes project using the analysis of connectomes pipeline. Then 871 quality MRI images is obtained by performing the pre-processing such as slice timing correction, normalization of voxel intensity and correction for motion with the phenotypic information. To delete fluctuations, nuisance regression method is used. By using the ABIDE and CC400 functional parcellation atlas of brain, the accuracy of 70.22% is obtained.

3   Methodology

3.1   Data Set Description

The experimental data set used in this paper is Autism Brain Imaging Data Exchange (ABIDE) publically available at the LONI website. The data has been collected at various levels such as functional and structural brain imaging data from various laboratories all around the world which help us advance our knowledge on the subject of neural bases of autism. This data set comprises of 1112 samples of which 539 individuals with autism spectrum disorder and 573 are typical controls.

In this study, a total of 233 images is used which comprises of 104 individuals with autism spectrum disorder and 129 typical controls. The image modality is magnetic resonance imaging (MRI). The image format is NIFTI. Each image comprises of three planes such as sagittal, coronal and axial. The research groups are autism and control patients. In radiology to obtain the pictures of anatomy and the physiological processes of our body, the clinical imaging technique called magnetic resonance
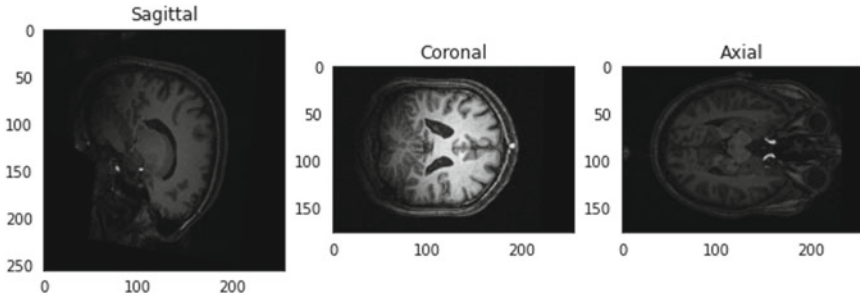
Fig. 1 MRI image

imaging (MRI) is used. The organs of the body are captured by using the strong magnetic field, radio waves and magnetic field gradients to create an image.

3.2 Image Description

The image comprises of three planes such as sagittal, coronal and axial. The image format is NIfTI, i.e. .nii file format. NIfTI is a type of file format for neuroimaging. It is an open file format which compress a huge matrix of elements and hence more suitable to store MRI images. To read the NIfTI file format image, the nibabel library is imported. Each image is comprises of different size. The total image size is 6 GB.

Figure 1 shows the MRI image with the shape of $176 \times 256 \times 256$. The 0th dimension of the image represents the sagittal view of the brain, 1st dimension of the image represents the coronal view of the brain, and the 2nd dimension of the image represents the axial or transverse view of the brain. As the images are of different size, it is difficult to process the image in convolutional neural network. Therefore, all the images are resized to the common size and then given to the convolutional neural network. Therefore, all images are resized to the shape of $160 \times 160 \times 160$.

3.3 Training and Testing Images

A total of 233 images were considered. In each image, the three respective channels were considered by splitting the images based on their planes such as sagittal, coronal and axial. Hence, a total of 699 images are obtained. The processing is done separately for all the planes individually and also finally done by using all the 699 images. Then, the total images were split into training and testing set of 559 images and 140 images, respectively. The training images were divided into batches of size 3. The network were randomly initialized and fine-tuned with AdaDelta optimization algorithm.
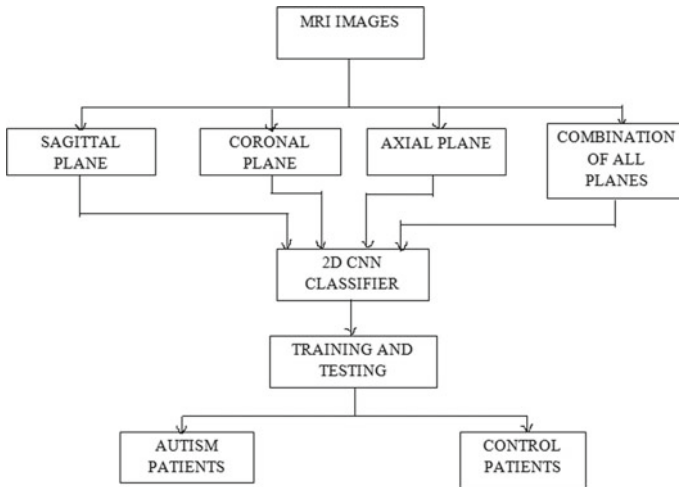
Fig. 2 Block diagram

The cross entropy loss were considered as a loss function and minimized using the optimization algorithm. The network was trained for 50 epochs.

3.4 Block Diagram

Figure 2 shows the block diagram that illustrates all the process involved. Initially, the input MRI image is split into sagittal, coronal and axial plane. Each plane is used as a separate data set for training and testing the model. Also, the model is tested and trained by the combination of all the planes. After forming the four different data set, it is passed into the 2D convolutional neural network for training and the testing process. The model predicts the image as either autism or control patients.

3.5 2D-CNN Classification

Figure 3 shows the CNN architecture used in the study for binary classification problem. The CNN contains three core layers such as convolutional layer, pooling layer and finally fully connected layer. The input image is passed as an input to the first convolutional layer with k filters and followed by a max pooling layer. The ConvNet architecture contains two such layers so as to keep less number of parameters in the network.

The network used a ReLU activation function in all the layers to avoid vanishing gradient problem, and all the parameters in the network were initialized with Xavier
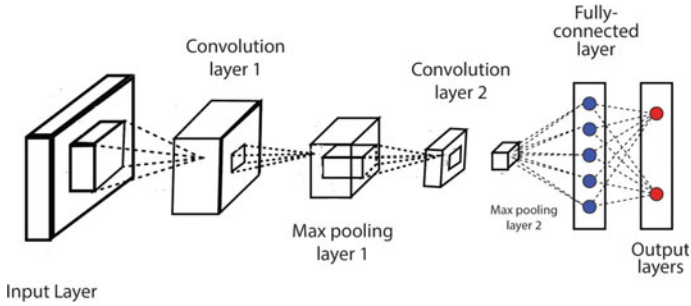
Fig. 3 CNN architecture

initialization technique. The final layer contains two neurons for each classes and uses softmax function to give the probability score. Cross entropy loss was used to optimize the network with AdaDelta optimizer.

4 Results and Discussion

The performance is evaluated for all the samples in the data set with four different strategies as mentioned above. The results are discussed in this section. In this, all the planes such as sagittal, coronal and axial planes are combined and given as a batch input to the network.

Figure 4 shows that the validation loss is almost similar as the training loss. It represents good fitting which means that the model is performed well on both the training data set and the validation data set.

Figure 5 shows that the training accuracy is similar as the validation accuracy. The model shows good accuracy in both the validation set and the training set. Therefore,

Fig. 4 Training and validation loss for the combination of all the planes
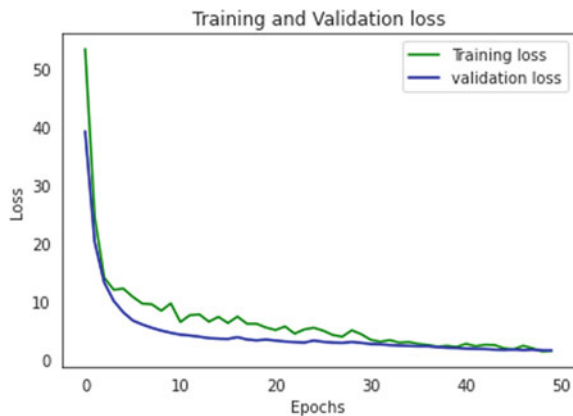
Fig. 5 Training and validation accuracy for the combination of all the planes

we can say that network avoids overfitting and underfitting problem and therefore generalizes well for the unknown samples. The overall performance of the network for all four planes is compared in Table 1 shown.

While the training and the validation is similar in the combination of all the planes followed by coronal plane, the difference is much higher in sagittal and axial plane.

Table 2 shows that the training accuracy is higher in combination of all planes followed by coronal, sagittal and axial plane. Also, the loss is higher in combination of all the planes.

Table 1 Overall training loss and training accuracy

Planes	Loss	Accuracy (%)	Validation loss	Validation accuracy (%)
Sagittal plane	0.5433	74.19	0.839	45.16
Coronal plane	0.9641	69.35	1.2925	58.06
Axial plane	0.8714	65.32	0.7378	50.00
Combination of all the planes	1.7217	62.61	1.5432	57.30
Sagittal plane	0.5433	74.19	0.839	45.16

Table 2 Overall testing loss and testing accuracy

Planes	Testing loss	Testing accuracy (%)
Sagittal plane	0.7407	55.31
Coronal plane	0.9640	61.70
Axial plane	0.7683	53.19
Combination of all the planes	1.3463	67.85

5 Conclusion

The proposed method helps significantly to classify the ASD patients from the control patients compared to existing approaches. The approach trained the CNN model using AdaDelta optimization algorithm and cross entropy loss and cast the problem as binary classification. From our study, we conclude that the training accuracy is higher in sagittal plane which is of 74.19%, but the validation accuracy is lower than the training accuracy. In case of combination of all planes and coronal plane, both the validation accuracy and training accuracy is similar, around 67%. Hence, we conclude that the model performs well on combination of all planes and coronal plane. In future, we like to classify the different levels in autism spectrum disorder such as mild, moderate and severe patients casting the problem as multi-class classification setting and changing the architecture accordingly.

References

1. Rathore A, Palande S, Anderson JS, Zielinski BA, Fletcher PT, Wang B (2019) Autism classification using topological features and deep learning. Natl Libr Med. Mostafa S, Tang L, Wu F-X (2019) Diagnosis of autism spectrum disorder based on eigen values of brain networks. IEEE Access 7
2. Anirudh R, Thiagarajan JJ (2019) Bootstrapping graph convolutional neural network for autism spectrum disorder classification. Lawrence Livermore National Laboratory, IEEE. Parikh MN, Li H, He L (2019) Enhancing diagnosis of autism with optimized machine learning models and persona characteristics data. Front Comput Neurosci 13
3. Heinsfeld AS, Franco AR, Craddock RC, Buchweitz A, Meneguzzi F (2018) Identification of autism spectrum disorder using deep learning and the ABIDE dataset. NeuroImage Clin 17
4. El Gazzar A, Cerliani L, van Wingen G, Thomas RM (2019) Simple 1-D convolutional networks for resting-state fMRI based classification in autism. In: International joint conference on neural networks. IEEE
5. Mellema C, Treacher A, Nguyen K, Montillo A (2019) Multiple deep learning architectures achieve superior performance diagnosing autism spectrum disorder using features previously extracted from structural and functional MRI. In: International symposium on biomedical imaging. IEEE
6. Sherkatghanad Z, Akhondzadeh M, Salari S, Zomorodi-Moghadam M, Abdar M, Rajendra Acharya U, Khosrowabadi R, Salari V (2020) Automated detection of autism spectrum disorder using a convolutional neural networks. Front Neurosci 13
7. Huang Z-A, Zhu Z, Yau CH, Tan KC (2020) Identify autism spectrum disorder from resting state fMRI using deep belief network. IEEE Trans Neural Netw Learn Syst
8. Ke F, Choi S, Kang YH, Cheon K-A, Lee SW (2020) Exploring the structural and strategic bases of autism spectrum disorders with deep learning. IEEE Access

Code Smell Detection Using Classification Approaches

Seema Dewangan and Rajwant Singh Rao

Abstract Code smells combine software features that indicate a code or design-related problem that can increase the difficulty of the software and make it complicated to recognize, develop, and maintain the software. Code smell detection helps us to make code accurate and effective. In this study, five classification models are applied. To complete this study, four datasets of code smell are used. To pick the finest feature in the existing dataset, a feature selection method (FST) is used. The ten-iteration cross-validation method is applied to calculate the execution of the model predicted. In this document, the random forest model for the Feature-envy dataset achieves an accuracy of 0.9912%.

Keywords Classification methods · Feature selection technique (FST) · Code smell detection

1 Introduction

The software complexity is continuously mounting because of the complex requirements, an increase in the quantity, module size, and code smells belonging to the advanced software, etc. Harsh conditions are challenging to examine and recognize, and therefore, as a result the improvement turns out to be problematic. The complicated necessities are not in the hands of the designer, but the smell of the code can be recognized and the software can be made modest, more comprehensible, and uncomplicated [1]. During the software making procedure, the operational and non-operational quality required for developers must be followed to secure software quality [2]. Developers concentrate only on practical demands and ignore non-functional needs, such as comprehensibility, verifiability, evolution, maintainability, and reusability [3]. The complexity of the software program is growing constantly due to the wide range of module sizes, method sizes, and branching complexity, renovation costs are increasing due to extra code smells, and the quality of the software

S. Dewangan (✉) · R. S. Rao
Guru Ghasidas Vishwavidyalaya (A Central University), Bilaspur, C.G., India
e-mail: sskd501@gmail.com

program is deteriorating due to the growing number of lines of code. Fowler et al. [4] described the refectory method from which the slackly executed program could be changed in an exemplary execution; 22 definitions of code smells were given by them.

Many approaches for code smell detection have been used in the literature to identify smells of different codes.

Every method will produce a distinct outcome according to their different categories. Seven types of code smell detection techniques are categorized by Kessentini et al. [5]. These seven techniques are cooperative-based technique [6], visualization-based technique [7], search-based method [8], probabilistic technique [9], symptom-based technique, metrics-based technique, and manual technique [10]. Inspection approaches [11], manufacturing process, and process identification methods [12] were used by manual technique to advance the superiority of software. To detect code smells using a symptom-based technique, the specification algorithm was used [13].

Multiple studies have analyzed the code smell's influence at the software and also displayed unwanted impacts of software's features with code smell detection [14, 15]. They also analyze the code smell effects that increase the risk of software system failures and faults. They found that the challenge full of code smell impacted the software improvement procedure and suggested software's refactoring for elimination of it.

Deligiannis et al. [16], Olbrich et al. [17, 18], and Khomh et al. [19] observed the effect of code smells on software development by observing the changes in the occurrence and size in the software system. They also observe that classes affected by code smells have a more significant rate of change and require supplementary maintenance work. The relevance of bad smells and the possibility of class error in an OOS system were investigated by Li and Shatnawi [20]. Infected software elements that use code smells have more class mistakes than other elements, according to the experiment. The negative effect of God-class on energy intake was examined by Perez-Castillo and Piattini [21], who found that eliminating God-class odors reduces the cyclomatic complexity of the source code in the software system.

The main contributions of this research work are divided into two parts: In the first part, five classification techniques are applied to detect the code smell from the dataset and feature selection technique is also applied to select the best features from each dataset. The second part shows the performance measures obtained using classification and evaluation techniques with the tenfold cross-validation technique.

In this research work, the classification techniques for code smell detection are proposed. The four datasets of code smell datasets are considered. The class-level smell contains the God-class and the Data-class datasets, whereas method-level smell contains Feature-envy and Long-method datasets. Five classification methods (random forest, SVM, naive Bayes, KNN, and logistic regression) are applied to classify the dataset.

This paper is partitioned in five section: Sect. 2 describes the literature review, and this section shortly reviews the work done in past by various authors for code

smell discovery by classification techniques. Section 3 describes proposed methodology. Section 4 describes the investigational consequences, and Sect. 5 explains the conclusion of our work.

2   Related Work

Many researchers have presented papers using ML algorithms to detect the code smells. In this paper, the existing methods of supervised learning techniques are used to detect code smells. In this paper, the existing methods of supervised learning techniques are used to detect code smells to comparing and experimenting with ML algorithms to detect code smells. They tested sixteen ML techniques on four code smell datasets as well seventy four Java platforms on the training dataset that were manually evaluated. Boosting approaches are also used on four datasets of code smells.

Mhawish and Gupta [22–24] proposed software metrics, tree-based and decision tree-based ML algorithms, and software metrics for differentiating and recognizing similar structural design patterns. To choose the most significant characteristics from each dataset, they utilized two feature selection strategies in light of GA-CFS (genetic algorithm).

They also employed a parameter refinement using a grid search method approach to improve the accurateness of all machine learning methods. Guggulothu and Moiz [25, 26] suggested a multi-label classification strategy for code smell detection. To see if the specified code components are affected in several ways, they employed a categorization system with many labels. For excellent accuracy, they made use of an unsupervised classification algorithm. Dewangan et al. [27] applied six ML algorithms, and two feature selection techniques such as chi-square and method for selecting features based on a wrapper were applied to pick the greatest features from each dataset; then, moreover grid search procedure was used to increase the performance of model, and they obtained 100% highest accuracy using the logistic regression technique for the Long-method dataset. Kreimer [28] proposed a detection approach to detect Long-methods and prominent class code smell based on a decision tree approach. The approach is evaluated on two small software: the WEKA software package and the IYC system. It was found that the prediction model and this model help detect code smells. The usefulness of decision trees for identifying code odors was proposed by Amorim et al. [29]. By putting Kreimer's decision tree model to the test, they were able to corroborate his findings. Class change proneness can be predicted based on code smell using Pritam et al.'s [30] machine learning methods. They agree that code smells have an influence on the predisposition of a given session in a produce context to change. They used six ML techniques to estimate variation proneness based on code smells from 8200 Java modules across 14 software systems.

Draz et al. [31] proposed employing the classifier based on the whale optimization method to enhance code smell prediction using a search-based method. They tested

Table 1 Summary of related work

Author name	Model	Datasets	FST	Results
Fontana et al. [33]	16 machine learning algorithm	Code smell datasets	Non	99.10% in B-J48 pruned on Long-method
Guggulothu and Moiz [25]	B-random forest, random forest, J48 unpruned etc.	Feature-envy and Long-method with multi-label classification	Non	95.9% in random forest on Long-method, 99.1% in B-J48 pruned on Feature-envy
Mhawish and Gupta [23]	Decision tree algorithm	Code smell dataset	Genetic algorithm-based two FST	Data-class—98.05%, God-class—97.56, and Long-method—98.38%
Mhawish and Gupta [22]	Six machine learning algorithm	Code smell dataset	Genetic algorithm-based two FST	99.70% in Data-class using random forest model

the nine different kinds of code smells on five different open systems applications. They had an accuracy of 94.24% and a recall of 93.4% on average.

For metric-based code smell detection, Pecorelli et al. [32] provided an interesting finding comparing the performance of machine learning-based and heuristic-based strategies. They considered five types of code smells (God-class, Spaghetti Code, Class Data Should Be Private, Complex Class, and Long-method) and compared ML techniques with DECOR, a state-of-the-art heuristic-based approach. Researchers discovered that the DECOR consistently outperformed the ML baseline. In Table 1, a summary table of some essential related work is shown.

3 Proposed Methodology

In this paper, a code smell detection framework is constructed using classification models. Code smell matrices play an essential part in determining the operational as well as non-operational abilities and recognizing the software's properties. Metrics manage the static information of the software, such as classes, methods, and parameters that measure coupling and cohesion between objects in the system. Figure 1 depicts the steps which are followed to build the code smell detection framework. First, four datasets of code smell are created. The pretreatment (regularization) processes are then done to the dataset to cover all of the dataset's ranges. The best features from each dataset are then selected using the wrapper-based FST. Then train the model with classification algorithms applied to the dataset and determine their performance. The methodology of tenfold cross-validation is then applied to assess the result of each experiment during the preparation development. For performance

Fig. 1 Proposed work

```
┌─────────────────────────────────────┐
│         Code Smell Dataset           │
└─────────────────────────────────────┘
                   │
                   ▼
┌─────────────────────────────────────┐
│       Normalization/ Preprocessing   │
└─────────────────────────────────────┘
                   │
                   ▼
┌─────────────────────────────────────┐
│       Feature selection technique    │
└─────────────────────────────────────┘
                   │
                   ▼
┌─────────────────────────────────────┐
│       Classification Algorithm       │
└─────────────────────────────────────┘
                   │
                   ▼
┌─────────────────────────────────────┐
│       10-Fold Cross-validation       │
└─────────────────────────────────────┘
                   │
                   ▼
┌─────────────────────────────────────┐
│         Final Evaluation             │
└─────────────────────────────────────┘
```
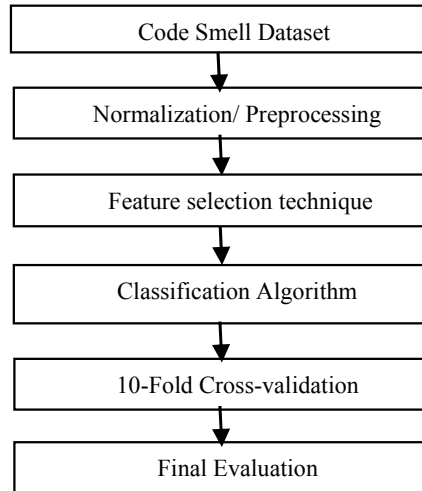
measurement, tenfold cross-validation is used, which divides the dataset into ten sections and repeats them ten times. Then evaluate the final results.

God-class, Data-class, Feature-envy, and Long-method are four code smell datasets which [33] were taken to make the code smell detection framework in this study. In the following section, the data preparation methodology is shown briefly.

Because various datasets have distinct attributes, we cannot always use straightforward classification techniques on them. As a result, normalization is required to span the dataset's various ranges. ML models may sometimes evaluate quickly on a normalized dataset, which might have a big impact when the model is sensitive to size. Prior to the implementation of the support vector machine algorithm, for example, it is necessary to avoid normalization in order to dominate higher number ranges on small number varieties, where the variety of possible elevated values causes mathematical problems [34]. This article uses the minimum–maximum normalization technique to convert dataset values between 0 and 1. This strategy is utilized in the data preparation step, which prepares the data for subsequent processing using one of the machine learning algorithms such as SVM, NN, and others [35]. The following equation executes an x mapping change from feature A from the range [min A, max A] to [new min A, new max A].

$$x' = (x - \min A)/(\max A - \min A)$$

All datasets were subjected to the min–max normalization approach, and the resulting new data was used as input into all classification systems.

To choose the best features (matrices) from each dataset, this experiment uses a wrapper-based FST. FST is applied to choose a set of characteristics in the dataset that are mainly appropriate to the goal value [36]. In this experiment, we have selected

the ten best features from each dataset, and then, classification algorithms are applied to each dataset.

This paper applies five classification algorithms (random forest, SVM, naive Bayes, KNN, and logistic regression) to perceive code smells from the code smell dataset. Classification algorithms classify the data into the specified number of classes in our dataset.

In this study, the validation technique is used to assess the performance of each experiment. For this, a tenfold cross-validation training approach was applied. Classification models that partition the dataset into tenfold with ten times of iteration are calculated using tenfold cross-validation. Different parts of the dataset are considered test datasets at each iteration, and other convolutions of the dataset are considered training modes. Then, finally the trained models are tested with unseen test dataset (10% split from the dataset before training). Stealth test dataset is used to clarify the model's forecasts and escape making broad generalizations.

Four performance constraints, precision, recall, $F1$-score, and accuracy, were examined to measure the efficiency of our classification approach. To calculate them, TP, TN, FP, and FN are found through the confusion matrix. True positive (TP) displays the occurrences in the positive class that properly forecast the model. False positive (FP) refers to occasions in which the model is predicted inaccurately in the positive class. True negative (TN) displays the instances in the negative class that properly forecast the model. Furthermore, false negative (FN) displays situations where the negative class is wrongly predicted.

4 Experimental Results

To the experiment work, four code smell datasets are used. The five classification algorithms are applied to identify the code smells from each dataset. The four performance measurements, precision (P), recall (R), $F1$-score ($F1$), and accuracy (A), are considered for each dataset. The experimental results for each classification technique are shown in Table 2. In this research, the $F1$-score was 0.98%, and accuracy was 0.98% for the Data-class, the $F1$-score was 0.98%, and the accuracy was 0.97% for the God-class, the $F1$-score was 0.98%, and an accuracy of 0.9912% for the Feature-envy, and an $F1$-score of 1.00% and an accuracy of 0.9952% for the Long-method using the random forest algorithm attained the maximum accuracy, whereas the naive Bayes (0.91% accuracy for Feature-envy) attained the gives the worst performance.

4.1 Evaluation of Our Techniques to Other Related Works

Table 3 represents a brief long evaluation of our techniques with other related works. In this evaluation, it is observed that in the Feature-envy dataset our approach achieved 99.12% accuracy, while in the Data-class and God-class datasets, Mhawish

Table 2 Experimental result of five classification techniques with four code smell datasets

	Data-class				God-class				Feature-envy				Long-method			
	P	R	F-1	A	P	R	F-1	A	P	R	F-1	A	P	R	F-1	A
Random forest	0.97	0.96	0.98	0.98	0.97	0.98	0.98	0.97	1.00	0.95	0.98	0.9912	1.00	1.00	1.00	0.9952
SVM	1.00	1.00	1.00	0.97	0.98	0.99	0.99	0.96	0.99	0.94	0.96	0.94	1.00	1.00	1.00	0.97
Naive Bayes	0.90	0.93	0.91	0.92	0.97	0.98	0.98	0.96	1.00	1.00	1.00	0.91	0.82	0.95	0.86	0.94
KNN	0.98	0.99	0.99	0.95	0.99	0.98	0.99	0.96	0.97	0.90	0.93	0.92	1.00	1.00	1.00	0.98
Logistic regression	0.98	1.00	0.99	0.96	1.00	0.98	0.99	0.96	1.00	0.97	0.99	0.96	1.00	1.00	1.00	0.98

P: Precision, R: Recall, F-1: F1-score, A: Accuracy

Table 3 Comparison of our approach with other related work

Year	Author name	Datasets			
		Data-class	God-class	Feature-envy	Long-method
		Accuracy (%)	Accuracy (%)	Accuracy (%)	Accuracy (%)
2016	Fontana et al. [33]	99.02	97.55	96.94	99.43
2020	Mhawish and Gupta [22]	99.70	98.48	97.97	95.97
2020	Guggulothu and Moiz [25]	N/A	N/A	99.10	95.90
2021	Dewangan et al. [27]	99.74	98.21	98.60	100.00
2021	Our approach	98.00	97.00	**99.12**	99.52

Bold value represent that our approach obtained the highest accuracy (99.12%) for feature-envy dataset in comparation to other authors

and Gupta [22] achieved 99.70 and 98.48% highest accuracy. For the Long-method data set, Dewangan et al. [27] achieved the highest accuracy of 100%.

5   Conclusion and Next Steps

The classification strategy is provided in this research to identify the code smells from software and to find the metrics that play an important part in the detection process using classification algorithms. To determine the key metrics that may be utilized to increase accuracy, the wrapper-based feature selection approach is used. The findings are then evaluated using a tenfold cross-validation procedure. In this research work, it has been noted that the random forest procedure achieved the maximum accuracy of 0.98% for Data-class, 0.97% for God-class, 0.9912% for Feature-envy, and 0.9952% for Long-method dataset. In the future work, other machine learning techniques and other metrics selection techniques can be applied to increase the outcomes.

References

1. Lehman MM (1980) Programs, life cycles, and laws of software evolution. Proc IEEE 68(9):1060–1076
2. Wiegers K, Beatty J (2013) Software requirements. Pearson Education
3. Chung L, do Prado Leite JCS (2009) On non-functional requirements in software engineering. In: Mylopoulos J, Borgida AT, Chaudhri V, Giorgini P, Yu E (eds) Conceptual modeling: foundations and applications-essays. Springer, pp 363–379
4. Fowler M, Beck K, Brant J, Opdyke W, Roberts D (1999) Refactoring: improving the design of existing code, 1st edn. Addison-Wesley Professional

5. Kessentini W, Kessentini M, Sahraoui H, Bechikh S, Ouni A (2014) A cooperative parallel search-based software engineering approach for code-smells detection. IEEE Trans Softw Eng 40(9):841–861

6. Abdelmoez W, Kosba E, Iesa AF (2014) Risk-based code smells detection tool. In: The international conference on computing technology and information management (ICCTIM2014). The Society of Digital Information and Wireless Communication, pp 148–159

7. Murphy-Hill E, Black AP (2010) An interactive ambient visualization for code smells. In: Proceedings of the 5th international symposium on software visualization. ACM, pp 5–14

8. Palomba F, Bavota G, Di Penta M, Oliveto R, Poshyvanyk D, De Lucia A (2015) Mining version histories for detecting code smells. IEEE Trans Softw Eng 41(5):462–489

9. Rao AA, Reddy KN (2007) Detecting bad smells in object oriented design using design change propagation probability matrix 1

10. Marinescu R (2004) Detection strategies: metrics-based rules for detecting design AWS. In: 20th IEEE international conference on software maintenance, 2004. Proceedings. IEEE, pp 350–359

11. Travassos G, Shull F, Fredericks M, Basili VR (1999) Detecting defects in object-oriented designs: using reading techniques to increase software quality. ACM Sigplan Not 34:47–56

12. Ciupke O (1999) Automatic detection of design problems in object-oriented reengineering. In: Technology of object-oriented languages and systems, 1999. TOOLS 30 proceedings. IEEE, pp 18–32

13. Moha N, Gueheneuc YG, Le Meur AF, Duchien L, Tiberghien A (2010) From a domain analysis to the specification and detection of code and design smells. Form Asp Comput 22(3–4):345–361

14. Yamashita A, Moonen L (2013) Exploring the impact of inter smell relations on software maintainability: an empirical study. In: Proceedings of the 35th international conference on software engineering, May 2013, pp 682–691

15. Yamashita A, Counsell S (2013) Code smells as system-level indicators of maintainability: an empirical study. J Syst Softw 86(10):2639–2653

16. Deligiannis I, Stamelos I, Angelis L, Roumeliotis M, Shepperd M (2004) A controlled experiment investigation of an object oriented design heuristic for maintainability. J Syst Softw 72(2):129–143

17. Olbrich S, Cruzes DS, Basili V, Zazworka N (2009) The evolution and impact of code smells: a case study of two open source systems. In: Proceedings of the 3rd international symposium on empirical software engineering and measurement, Oct 2009, pp 390–400

18. Olbrich SM, Cruzes DS, Sjøberg DIK (2010) Are all code smells harmful? A study of god classes and brain classes in the evolution of three open source systems. In: Proceedings of the 26th IEEE international conference on software maintenance, Sept 2010

19. Khomh F, Penta DM, Gueheneuc YG (2009) An exploratory study of the impact of code smells on software change proneness. In: Proceedings of the 16th working conference on reverse engineering, Oct 2009, pp 75–84

20. Li W, Shatnawi R (2007) An empirical study of the bad smells and class error probability in the post-release object-oriented system evolution. J Syst Softw 80(7):1120–1128

21. Perez-Castillo R, Piattini M (2014) Analyzing the harmful effect of god class refactoring on power consumption. IEEE Softw 31(3):48–55

22. Mhawish MY, Gupta M (2020) Predicting code smells and analysis of predictions: using machine learning techniques and software metrics. J Comput Sci Technol 35(6):1428–1445. https://doi.org/10.1007/s11390-020-0323-7

23. Mhawish MY, Gupta M (2020) Software metrics and tree-based machine learning algorithms for distinguishing and detecting similar structure design patterns. SN Appl Sci 2:11

24. Mhawish MY, Gupta M (2019) Generating code-smell prediction rules using decision tree algorithm and software metrics. Int J Comput Sci Eng (IJCSE) 7(5):41–48

25. Guggulothu T, Moiz SA (2020) Code smell detection using multi-label classification approach. Softw Qual J 28:1063–1086. https://doi.org/10.1007/s11219-020-09498-y

26. Guggulothu T, Moiz SA (2019) Detection of shotgun surgery and message chain code smells using machine learning techniques. Int J Rough Sets Data Anal (IJRSDA) 6(2):34–50. https://doi.org/10.4018/IJRSDA.2019040103
27. Dewangan S, Rao RS, Mishra A, Gupta M (2021) A novel approach for code smell detection: an empirical study. IEEE Access 9:162869–162883. https://doi.org/10.1109/ACCESS.2021.3133810
28. Kreimer J (2005) Adaptive detection of design flaws. Electron Notes Theor Comput Sci 141(4):117–136
29. Amorim L, Costa E, Antunes N, Fonseca B, Ribeiro M (2015) Experience report: evaluating the effectiveness of decision trees for detecting code smells. In: Proceedings of the 26th IEEE international symposium on software reliability engineering, Nov 2015, pp 261–269
30. Pritam N, Khari M, Son LH, Kumar R, Jha S, Priyadarshini I, Abdel-Basset M, Long HV (2019) Assessment of code smell for predicting class change proneness using machine learning. IEEE Access 7:37414–37425
31. Draz MM, Farhan MS, Abdulkader SN, Gafar MG (2021) Code smell detection using whale optimization algorithm. Comput Mater Contin 68(2):1919–1935
32. Pecorelli F, Palomba F, Di Nucci D, De Lucia A (2019) Comparing heuristic and machine learning approaches for metric-based code smell detection. In: 2019 IEEE/ACM 27th international conference on program comprehension (ICPC), pp 93–104. https://doi.org/10.1109/ICPC.2019.00023
33. Fontana FA, Mäntylä MV, Zanoni M, Marino A (2016) Comparing and experimenting machine learning techniques for code smell detection. Empir Softw Eng 21(3):1143–1191
34. Forman G, Scholz M, Rajaram S (2009) Feature shaping for linear SVM classifiers. In: Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining, June 2009, pp 299–308
35. Ali PJM, Faraj RH (2014) Data normalization and standardization: a technical report. Mach Learn Techn Rep 1(1):1–6
36. Soto AJ, Cecchini RL, Vazquez GE, Ponzoni I (2008) A wrapper-based feature selection method for ADMET prediction using evolutionary computing. In: Marchiori E, Moore JH (eds) Evolutionary computation, machine learning and data mining in bioinformatics. EvoBIO 2008. Lecture notes in computer science, vol 4973. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-78757-0_17

An Approach Toward Design and Implementation of Distributed Framework for Astronomical Big Data Processing

R. Monisha, Snigdha Sen, Rajat U. Davangeri, K. S. Sri Lakshmi, and Sourav Dey

Abstract Due to advancement of modern technology, data generation is becoming huge in all sectors in recent times. The observational astronomy has embraced modern tools, thereby generating large data. Analyzing and extracting useful pattern from those data is the need of the hour. In this paper, we have tried to implement several machine learning algorithms using Apache Spark to process this massive amount of data. The case study from cosmology we considered here is photometric redshift estimation which is a dominant research area in astronomy. Due to high end telescopic camera, lot of astronomical data is being generated which need to be analyzed efficiently and quickly. In this work, we have implemented Artificial Neural network (ANN), Random Forest, Linear Regression, and Decision Tree algorithm on Apache Spark to predict redshift of galaxies and quasars. The focus area of our study is to explore and compare execution time of those four machine learning algorithms and provide a detailed study of their performance in distributed environment as well as standalone system. The dataset used here are collected from Sloan digital Sky survey (SDSS) which is a wide range in depth sky survey. Our work shows that Random Forest outperforms other algorithms in terms of predictive performance in both the environments. Although we experimented on subset of data, scalability issue also can be treated using big data framework.

Keywords Astronomical big data · Distributed environment · Spark · Machine learning

1 Introduction

Machine learning (ML) is a branch of Artificial Intelligence (AI) and computer science which focuses on the use of data and algorithms. These algorithms and modules subsequently drive decision-making within applications and businesses, ideally impacting key growth metrics. In addition to rapid data surge in astronomy,

R. Monisha · S. Sen (✉) · R. U. Davangeri · K. S. Sri Lakshmi · S. Dey
Department of CSE, Global Academy of Technology, Bengaluru, Karnataka, India
e-mail: snigdha.sen@gat.ac.in

the nature of data is high dimensional and more complex. Traditional data processing approaches are no more efficient enough to carry out this task. Therefore, a huge demand to understand and analyze these data precisely and accurately arises that subsequently lead to unknown discoveries.

Big data concepts and machine learning together have proven to be the efficient tool in this regard. The main aim of our project is to explore the efficacy of multiple ML algorithm in distributed environment. Our project is based on the distributed system which makes ML tasks of big data scalable, flexible, and efficient. The most interesting aspect of the distributed system is distribution of workload in multiple nodes. It emphasizes on moving the computation to the data rather than moving the data to the computation. The distributed big data processing framework Apache Spark utilizes parallel computing concepts to distribute data into multiple worker nodes and facilitates quicker processing.

Astronomical datasets are typically very large. The Sloan Digital Sky Survey (SDSS) is a large collection of astronomical data which comprises the most detailed three-dimensional maps of the Universe, and a major multi-spectral imaging and spectroscopic redshift survey. Redshift is a crucial parameter through which we find out the distance of extragalactic objects from earth. Hence, prediction of redshift is an essential task in the context of cosmology. The issues with processing astronomical data are to classify accurately and efficiently observed celestial objects.

Analyzing the SDSS data, using the most efficient algorithms has been proved that it would consume more amount of time. Most of the astronomical operations are very expensive to process of the vast feature of data, and these methodologies can be taken care by using certain specified big data properties. There are a million of objects in the photometric catalog which maintains physical properties of the galaxies. Apart from that, the cross-validation of astronomical catalogs from a collection of well-versed solution in terms of their information and statistical recognition of the objects helps for the advancement of the computer technology in the Astro-physics field.

To resolve these difficulties, a general admitted solution is to use data partitioning to parallelize query computation by distributing the data on different worker nodes. The primary aspect of our paper is to propose an approach using Apache Spark that can handle and process this huge amount of data precisely and efficiently.

Further organization of the paper is like below. Section 2 describes existing literature studies, and Sect. 3 presents description of the dataset. In Sect. 4, we illustrate about our methodology and experimental setup. Results and discussion section have been described in Sect. 5. Finally, we conclude with prospective future enhancements.

2 Literature Survey

Despite many researchers have experimented with various models to classify and analyze redshift dataset over the years, we cite few relevant approaches here. In recent years, deep learning [1] has shown high improvements in classifying images of galaxies. In the paper [2], Kasun has exposed the usage of Hadoop, and map-reduce

technique hypothesis generation and data-driven modeling are used to understand behavior and interactions. They explored in detail the nature of Astronomical Data, applications of big data in Astronomy, Large scale data analysis in Astronomy, and Digital Sky surveys. The author believes that a large amount of data in Terabytes are being produced by various ongoing sky surveys ranging from different wavelengths.

In the paper [3], authors have adapted different technologies that have driven the progress in the astronomy. These technologies have come up with the development of astrophotography from the survey perspective. The technologies that work together with Big Data such as Hadoop, HDFS, Cassandra, Mango DB, NoSQL, and MapReduce have intensions to achieve goal in extracting the patterns from existing data.

To classify the entire SDSS-III photometric data into stars, quasars, and galaxies. Acharya et al. [4] set up a cloud environment and apply different machine learning classification algorithms to choose the best among them, which can classify the entire data precisely abiding by the evaluation metrics, within a reasonable amount of time. In [5], authors specifically explained the reason for the selection of the random forest algorithm which ensembles the Decision Tree. According to the author, the dataset must be broken down into smaller subsets, which is associated with a Decision Tree which is incrementally developed. At the end, a tree with leaf node and decision node is available. In the final prediction particularly for SDSS data, 90% of the objects are non-quasars. So, any classifier which would classify entire data into non-quasar would get an accuracy of 90% which seems impressive.

In another research work, astronomical data mining tool CANFAR + Skytree [6] was proposed by Ball which is considered first cloud project on astronomy. Very recently, a Spark-based framework was proposed by Hong et al. [7] for understanding topological structure of gravitational lensing and clustering.

Furthermore, research by Brahem et al. [3] focused on implementing Astroide software which provides an in memory solution for query, optimization, and astronomical big data processing. Image processing framework Kira was proposed by Zhang et al. [8] to give more robust performance improvement over supercomputer. In memory computation and real time data processing capabilities of facility Spark facilitates faster and efficient data analysis. In recent times, Sen et al. [9] has implemented neural network model using Apache Spark on AWS cloud platform for prediction of redshift.

3 Dataset Description

The dataset used here is acquired from SDSS [10, 11] which contains photometric and spectroscopic catalogs. Through these catalogs, we collected required dataset to perform redshift estimation task. Our dataset contains approx. 25 lakhs records of celestial objects. This dataset consists of 26 columns containing model magnitude, fiber magnitude, Petrosian flux, band overlapping magnitude and many other properties of galaxy and quasars. First 25 features in the dataset are considered as input

features and 26th column is the redshift which is a target variable, and we aim to estimate that in our study. Greater the redshift [12] value, greater is the distance from earth. This redshift value is measured by shooting a light from earth to other galaxies, when the light travel from earth to galaxies the amount of light shifted toward red spectrum is called the redshift.

4 Methodology and Experimental Setup

The entire experiment has been conducted in Google Collaboratory [13] using PySpark. Data analysis, model development, and data visualization have been done using multiple python module such as scikit learn, pandas, NumPy, seaborn, Matplotlib, etc. PySpark [14] is a python-based API which is useful to work in Spark environment using python. Furthermore, Spark [15] is a real time big data processing framework which offers in memory processing. It is considered to be much faster than Hadoop MapReduce. Matplotlib and seaborn was used for data visualization. After building the model, we run it on Spark to leverage its distributed analytic engine. The usage of machine learning algorithms over the Spark framework to predict redshift is an important aspect of our work.

As there is no library for neural network in Spark, we used Elephas [16] which is an extension to keras that can deal massive data in Spark. Among two different approach of data parallel and model parallelism, we consider data parallel strategy here, where entire dataset will be distributed into multiple nodes. This kind of distribution method has massive impact in reducing execution time. We create an estimator like some of the PySpark pipeline and set the optimizer settings and then pass that to our Elephas estimator. Figure 1 shows the working of Elephas library, and Fig. 2 discusses the overall flow of our proposed approach. We have used Adam optimizer for optimizing our ANN model.

Fig. 1 The working of Elephas
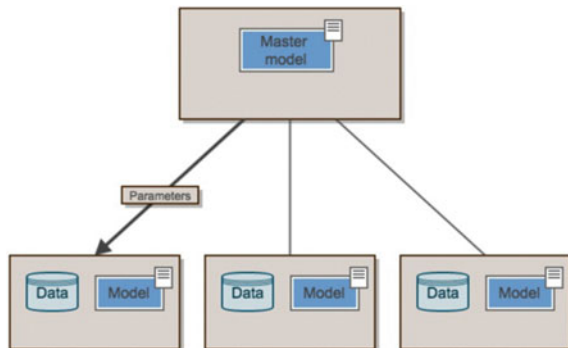
Fig. 2 Proposed workflows
of our system
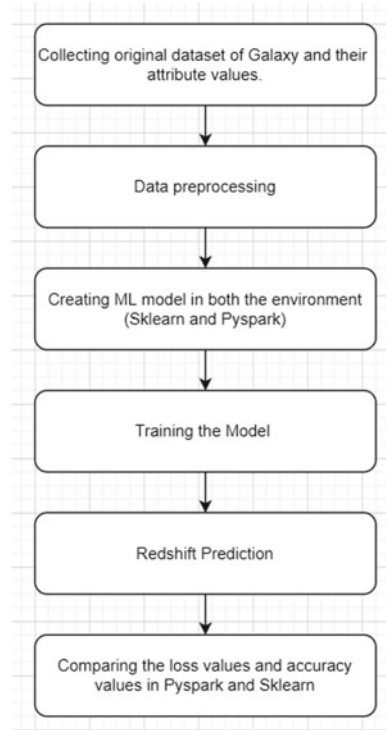
4.1 Processing of Data

Data preprocessing is an essential task for qualitative predictive power of a model. Our dataset contains all numeric values in all the attributes. We have preprocessed the data by eliminating the negative infinite values (in our case -9999) and null values. Number of missing values identified was less, therefore, removal of those value did not cause any significant difference. The dataset is further preprocessed by removing unwanted columns using correlation values of every column with the target column through a heat map. After feature reduction, we are left with 22 features and 1 target column. Figure 3 shows the heat map representation of all 26 variables in the dataset.

The main purpose of generating a heat map here is to show all the feature variables as row headers and column headers which is an extremely powerful way to visualize relationships between variables in high dimensional space. The values close to 1 are more correlated with target variable.
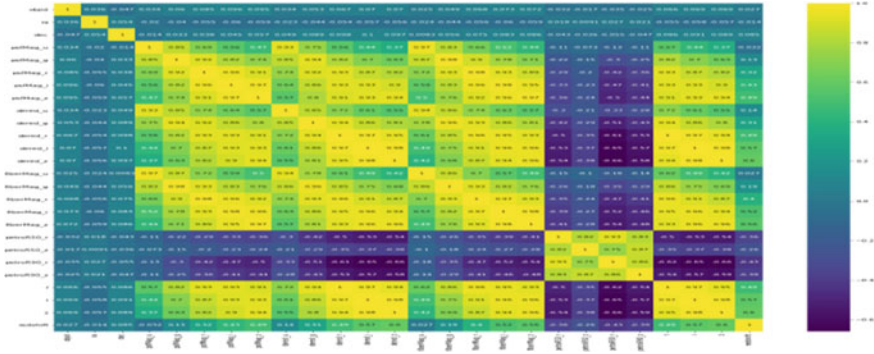
Fig. 3 The plot for heat map of correlation values

4.2 Preparation for Data Modeling

Once we have identified the features and target value in our dataset, we divided our dataset as 80% for training and 20% for testing. Feature scaling has been applied on training dataset before fitting it to the model. In our experiment, we converted all input features in [0, 1] range. After this step, ML models like ANN and other such as Linear Regression, Decision Tress, and Random Forest are directly trained using training dataset. Training time of model execution and Error metrics like Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R^2 score is noted down for comparing the performance of these algorithms. In Table 1, we present the performance measurement of these four algorithms. The training and testing are done in two different environments (Sklearn and PySpark).

5 Results and Discussion

Comparative analysis of ML algorithm: We have used the most significant supervised algorithms like Linear regression, Decision Tree, Random Forest, and ANN. The plot of actual redshift versus predicted redshift of all the algorithms in two different environments is shown in Figs. 4 and 5.

From Table 1, it is visible that performance of Random Forest algorithm in Spark as well as Sklearn is much better than other all algorithms. In Table 2, we discuss about training time in both environments.

From Table 2, it is evident that most of the algorithm executed in PySpark in less time than Sklearn environment as it provides the feature of distributed computing.

Data analysis and visualization: Here in Fig. 6, we have shown a bar plot representing the time difference between the distributed environment and single machine. From bar chart and Table 2, we can see that time of execution in Sklearn is lot more that
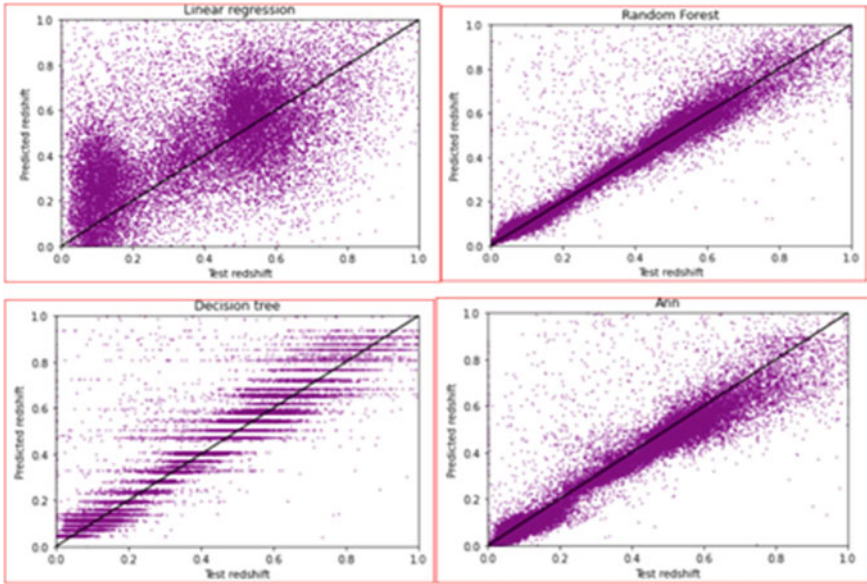
Fig. 4 Plot graph for predicted versus actual redshift in Sklearn
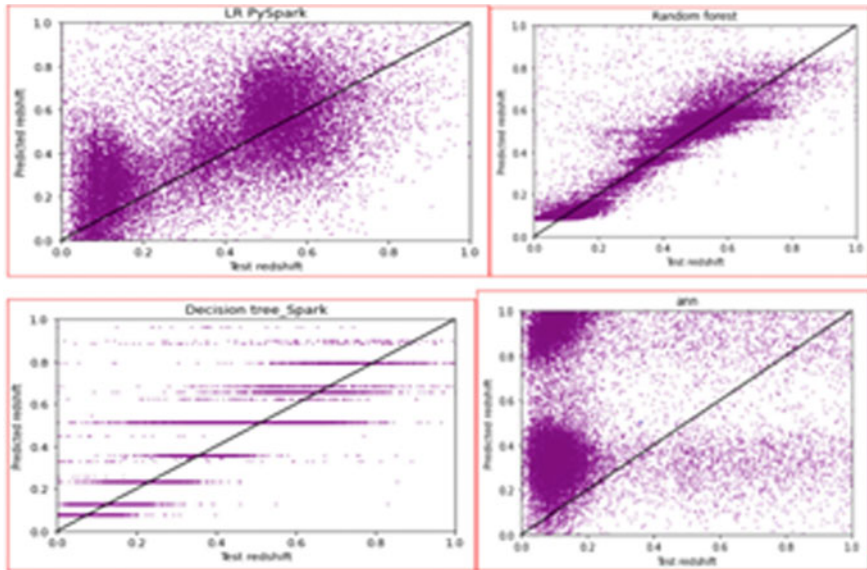
Fig. 5 Plot graph for predicted versus actual redshift using PySpark

Table 1 Comparative study of RMSE, R^2, MAE, MSE error of all 4 algorithms

	Sklearn	Spark		Sklearn	Spark
Decision Tree			*ANN*		
RMSE	0.54	0.455	RMSE	0.4	1.13
R^2	0.65	0.60	R^2	0.68	−0.21
MAE	0.19	0.212	MAE	0.167	1.6
MSE	0.183	0.24	MSE	0.165	0.43
Linear regression (LR)			*Random forest*		
RMSE	0.495	0.494	RMSE	0.395	0.45
R^2	0.53	0.541	R^2	0.70	0.61
MAE	0.295	0.295	MAE	0.14	0.208
MSE	0.245	0.247	MSE	0.156	0.21

Table 2 Comparison of training time in Sklearn and PySpark

Algorithm	Execution time in Sklearn in min	Execution time in PySpark in min
Linear regression	0.046	0.0058
Random forest	21.2	6.48
Decision Tree	1.5	0.048
ANN	68.82	0.11

Fig. 6 Visualization of training time of all algorithms using Matplotlib
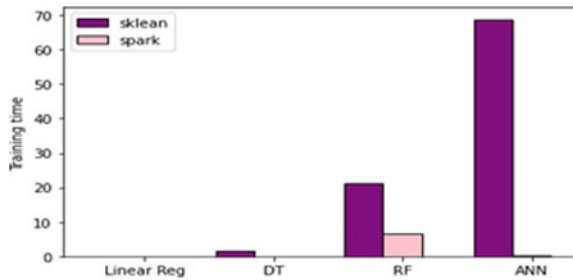
of PySpark [7]. Time of execution for all algorithms is a lot less in PySpark, which a big advantage when we try to execute any neural network algorithm like ANN using massive amount of data.

6 Conclusion

In this paper, we showed the efficacy of distributed environment to process large amount of data. Using Decision Tree and ANN, one can analyze a data set of millions

of galaxies in just a few hundred seconds using PySpark. Spark being a distributed framework, it is seen that the execution time is faster though it evaluates using a lazy approach compared to Sklearn. Apache Spark and Elephas library collectively are proven to be a powerful tool in distributed machine learning approach while giving improvement in performance. The accuracy and time efficiency are the main aspect of this project, and it can be concluded that using a distributed environment the execution time is very less compared to that of a standalone environment, and Random Forest is the most efficient among all four algorithms for this task. As future studies, we plan to work with other sky survey dataset and explore the efficiency of our model.

References

1. Sandeep VY, Sen S, Santosh K (2021) Analyzing and Processing of Astronomical Images using Deep Learning Techniques. 2021 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT). IEEE
2. Kasun Chathuranga Pererea KL (2018) Big data in astronomy. https://doi.org/10.13140/RG.2.2.31794.96962
3. Brahem M, Zeitouni K, Yeh L (2018) ASTROIDE: a unified astronomical big data processing engine over Spark. IEEE Trans Big Data 6(3):477–491
4. Acharya V, Bora PS, Navin K, Nazareth A, Anusha PS, Rao S (2018) Classification of SDSS photometric data using machine learning on a cloud. IEEE 115(2):249–257
5. Zhang Y, Zhao Y (2015) Astronomy in the big data era. Data Sci J 14(11):1–9
6. Ball NM (2013) CANFAR + Skytree: a cloud computing and data mining system for astronomy. arXiv preprint arXiv:1312.3996
7. Hong S et al (2020) Constraining cosmology with big data statistics of cosmological graphs. Mon Not R Astron Soc 493(4):5972–5986
8. Zhang Z et al (2016) Kira: processing astronomy imagery using big data technology. IEEE Trans Big Data 6(2):369–381
9. Sen S et al (2021) Implementation of neural network regression model for faster redshift analysis on cloud-based Spark platform. In: International conference on industrial, engineering and other applications of applied intelligent systems. Springer, Cham
10. https://www.sdss.org/
11. Vanzella E, Cristiani S, Fontana A, Nonino M, Arnouts S, Giallongo E, Grazian A, Fasano G, Popesso P, Saracco P, Zaggia S (2004) Photometric redshifts with the multilayer perceptron neural network: application to the HDF-S and SDSS. A&A 423:761–776
12. Sen S et al (2022) Astronomical big data processing using machine learning: A comprehensive review." Experimental Astronomy 1–43
13. https://colab.research.google.com/
14. http://Spark.apache.org/docs/latest/api/python/
15. https://Spark.apache.org/
16. Elephas homepage. https://github.com/maxpumperla/elephas

On Embedding Properties of Double-Star Interconnection Network Topology

Nirmal Keshari Swain, Chinmaya Ranjan Padhan, and Nibedita Adhikari

Abstract The double-star (DS) interconnection network is a derived network. It is a hybrid network with the *n*-star as its basic building block. It is having both degree and diameter *n* where *n* is the network dimension similar to the hypercube structure. In comparison to other derived networks, the double-star network retains all the topological properties of the base or parent network. For the study of the robustness of this network, the embedding of ring and mesh networks into the double star is studied. Also, the Hamiltonian properties are satisfied by the proposed DS network.

Keywords Embedding · Star · Ring · Mesh

1 Introduction

Parallel processing algorithms and architectures have been studied for decades as a way of improving computer system performance [1]. Parallel computing is a type of computational system in which both calculations and message passing among processes are carried out simultaneously. Now, massive parallel computing is done at the background of every scenario including big data. Here, a problem of large size is divided into multiple smaller and similar parts that can be processed simultaneously by multiple processors. Thus, parallel computing enhances the processing speed of a system. For this type of processing, multiple instructions, multiple data (MIMD) computers are the most popular concept [2, 3]. And to implement MIMD, interconnection networks are needed. In a parallel interconnection network, a machine can send information from the source node to the destination node which should be completed with as minimum possible latency. As compared to other machine, PIN is less expensive. In literature, there are several types of PIN existing, and hypercube

N. K. Swain (✉)
Gandhi Institute for Education and Technology, Baniatangi, Khurda, Odisha, India
e-mail: swain.nirmal6@gmail.com

C. R. Padhan · N. Adhikari
Centre for Advanced Post Graduate Studies, Biju Patnaik University of Technology, Rourkela, Odisha, India

and star graphs are the most popular [3–5]. In due course of time, the basic networks are modified to improve their topological parameters. The derived networks either show better node capacity or faster message passing [6–10]. Both these properties make the PINs a basic or mandatory need for a big data system. Now a days, big data is a very popular buzzword and attracting huge attention. This current work is inspired from Double Star (DS) network [9].

A double-star network consists of two-star graphs combined into one, each one of $n!$ nodes. In a DS, all nodes are of the same degree n. Due to this property, more nodes can be accommodated into the network at a low-node degree. The embedding properties of the interconnection networks play an avital role as it emphasizes the message passing. The broadcasting process involves finding a spanning broadcasting tree inside the network. The height of this tree tells about the time consumed in message passing from one node to rest. The height of this spanning tree is equal to the diameter of the network. Similarly, the embedding of other networks like ring and meshes also reveal the robustness of the host network. With this embedding, we are attempting to make this parallel interconnection network a faster and reliable message passing system for big data processing.

The rest part of the paper is organized as follows. Section 2 describes the background of the current work. Section 3 describes the Hamiltonian properties in the DS network. Section 4 describes the embedding properties of a double star. Finally, Sect. 5 concludes the paper.

2 Background

The n-dimensional star is called n-star (S_n) is shown in Fig. 1. It is a node-symmetric graph consisting of $n!$ nodes and an edge-symmetric graph having $\frac{n!(n-1)}{2}$ number of edges. The nodes are represented by permutations of n symbols. Each vertex in a n-star has $(n-1)$ incident edges. It is very much hierarchical. For the same number of nodes, it has a lower node degree and diameter in comparison to the hypercube. Thus, it appeared as an attractive alternative to the n-cube. However, it is not a regular graph like the n-cube. Also, expansion from S_n to S_{n+1} is not easy as several nodes

Fig. 1 Star graph S_3
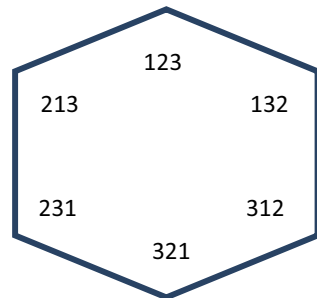
Fig. 2 Double-star graph, DS$_3$
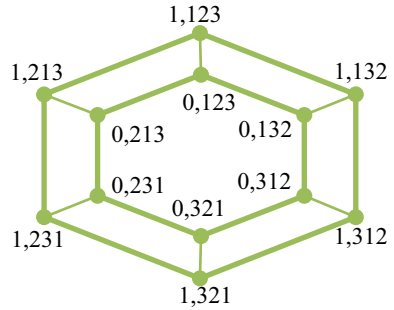
are added. There are various derivatives of star graph already proposed, namely star varietal cube [6], meta-star [7], and star-crossed cube [8] to overcome the node packing problem in the network.

Recently, the double star (DS) is proposed as a derived network. Double-star graph is designed using the properties of star graph, and it retains the originality of the parent structure. Here, edges of two-star nodes are connected. Figure 2 shows the interconnect of nodes of a DS graph. The n-dimensional double-star network is called as DS$_n$. DS network holds all the topological properties of the star interconnection network. As with the star network, in DS, ring and mesh property can be embedded easily. In this paper, different topological parameters are discussed. All the properties of the network are already proved in our previous paper [9]. Also, load balancing properties are studied to establish the superiority of the DS [10].

Theorem 1 *The number of nodes in the DS$_n$ graph is $2 \times n!$*

Theorem 2 *Total number of edges in the DS$_n$ graph is $E = n \times n!$*

Theorem 3 *The degree of each node of the DS$_n$ is n.*

Theorem 4 *The diameter of the Double Star is $[3(n-1)] + 1$.*

Theorem 5 *The cost of the DS$_n$ network is $n \times ([3(n-1)] + 1)$.*

The double-star graph is regular as the degree and diameter of the network are the same as the hypercube structure [5]. The rings are of practical importance as they resemble many real-world problems. Many computational problems in parallel processing can be regarded as graph embedding problems and solved easily. For ring embedding, Hamiltonicity is a sufficient and necessary condition. Hence, the next section discusses the Hamiltonicity of the DS network.

3 Hamiltonian Property of Proposed Network DS

A Hamiltonian cycle, a kind of closed graph that visits every node of the graph exactly once. It was named after Sir William Rowan Hamilton [11, 12]. In this path, a certain label is assigned to each node. The label of a node depends upon the position in the Hamiltonian path. Label 1 is assigned to the first node where the last node is assigned as $P - 1$. Here, P is the total number of nodes in the particular network. It has many applications in network topology, time scheduling, and designing fault-tolerant routing techniques. The Hamiltonian path constructed in the DS_4 network is shown in Fig. 3.

Theorem 6 *Double star (DS_n) contains a Hamiltonian cycle.*

Proof A star graph contains multiple Hamiltonian cycles. Finding the Hamiltonian cycle of a graph is an NP-complete problem. However, the existence of the Hamiltonian cycle in a graph had been proved previously [13, 14]. As per the conditions given by Mehedy et al. [15] in 2007, there must be a Hamiltonian cycle if the graph contains at least $\frac{n^2}{4}$ edges [10]. We know that the number of edges E of DS_n is $(n \times n!)$. To prove the existence of the Hamiltonian cycle in the DS_n graph, total edges must be more than $\frac{n^2}{4}$ for $n > 1$.

Now, we can prove this equation by the mathematical induction method.

Let, $f(n) = n \times n!$

For $n = 2$, $f(n)$ is 4.

So, $f(n) > \frac{n^2}{4}$ is true for $n = 2$.

Similarly, assume that $f(n)$ is true for $n = k$.

So, $k \times k! > \frac{k^2}{4}$.

Now, for $n = (k + 1)$, we have to prove that

$$f(k + 1) = (k + 1) \times (k + 1)! > \frac{(k + 1)^2}{4}$$

Consider, $f(k + 1) = (k + 1) \times (k + 1)!$

Fig. 3 Hamiltonian path in DS_3
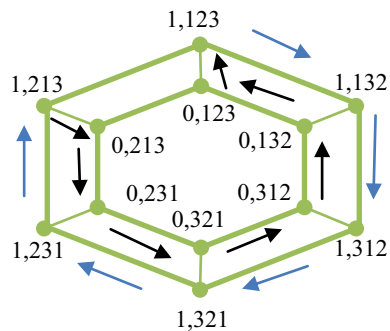
$$= (k + 1) \times (k + 1) \times k!$$

$$= (k^2 + 2k + 1) \times k!$$

$$= k(k \times k!) + 2(k \times k!) + k!$$

$$= k \times f(k) + 2 \times f(k) + k! = (k + 2)f(k) + k!$$

$$> (k + 2) \times \frac{k^2}{4} + k! > \frac{k^2}{4} + k!$$

$$> \frac{k^2 + 4k!}{4} > \frac{k^2 + 4k + 4(k + 1)}{4}$$

$$> \frac{k^2 + 2k + 1}{4} = \frac{(k + 1)^2}{4}$$

$$\Rightarrow f(k + 1) > \frac{(k + 1)^2}{4}$$

So, the statement is true for $n = k + 1$

$$\Rightarrow f(n) = n \times n! > \frac{n^2}{4}$$

Hence, the DS contains a Hamiltonian cycle. (Proved)

4 Embedding in DS

The embedding of different topological parameters is considered to be an important property of any topology. The embedding of topologies like rings, meshes, and binary trees into star graphs, CQs, and SCQ is already previously studied. Many researchers have emphasized the importance of embedding into a host network. Dilation, congestion, and expansion are the measures to guide the quality of embedding. Two types of embedding are explored in the DS here. The following section describes the embedding possibilities of rings and meshes into the DS network.

The embedding of a graph $G(V, E)$ into another graph $S(V, E)$ is a mapping of the set of vertices $V(G)$ into that of S that is $V(S) \cup E(G) = E(S)$. R denotes the mapping. Thus, any vertex X of G is mapped through $R(x)$ of S uniquely. The congestion of any edge e of S is the number of paths (each path representing an edge

of G mapped to S) of G, which contains e. The maximum of the congestions of all of the edges of S is the congestion of the embedding. The dilation is the maximum distance between the images of two adjacent nodes of guest graph G in the host graph S. Last but not least, the expansion is the ratio of the cardinality of the set of vertices of both the networks.

4.1 Ring Embedding in DS

DS is an interconnection of two multiple stars connected. So, like a star network, we can derive ring properties in a DS. Generally, a ring refers to the traveling path where starting and end nodes are identical irrespective of traveling the same node multiple times. This Hamiltonian property can be derived in a DS. Here, all the nodes of the DS network must be traveled for a single time and finally, the path ends by reaching the node from which the path was started.

Lemma 1 *Ring embedding is possible in DS(3), and the maximum size of the ring is 12.*

A ring can be embedded into the DS_3 using the gray-code concept. A gray code is a well-known sequence of binary bits where two consecutive codes differ by only 1 bit $G_1 = (0, 1)$. G_2 can be derived as $(00, 01, 11, 10)$. For $n > 2$ $G_n = (0G_n - 1, 1G_n^r - 1)$ where G_n^r is the reverse string of G_n hence $G_3 = (0G_2, 1G_2^r) = (000, 001, 011, 010, 110, 111, 101, 100)$. A ring with 12 nodes R_1 to R_{12} bearing the node addresses of the DS graph is as follows, and the ring network can be shown as Fig. 4.

Fig. 4 Ring embedding in DS_3
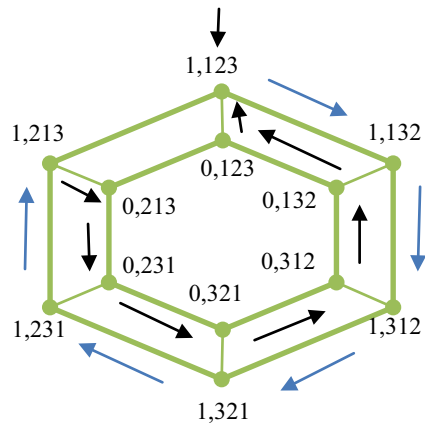
$$R_1 = (1.123) \qquad R_2 = (1.132) \qquad R_3 = (1.312) \qquad R_4 = (1.321)$$
$$R_5 = (1.231) \qquad R_6 = (1.213) \qquad R_7 = (0.213) \qquad R_8 = (0.231)$$
$$R_9 = (0.321) \qquad R_{10} = (0.312) \qquad R_{11} = (0.132) \qquad R_{12} = (0.123)$$

Here, to get a ring in a DS_3 network, 12 edges are to be considered. So, the total number of edges of the DS_3 ring is $E = 2 \times n!$

Lemma 2 *Ring embedding is also possible in DS(4), and the maximum size of the ring is* $2 \times n! = 48$.

In the DS_4 network, four DS_3 networks are connected bearing the properties of a star. To create a ring in DS_4, all the nodes in the network will be traveled only once, and the starting and ending node must be identical. After traveling one cluster, one of the next connected clusters will be traveled as shown in Fig. 5.

DS_4 network contains $2 \times n! = 48$ number of nodes and 96 edges as shown in Fig. 5, and all are connected. So, the degree of each node in DS_4 is 4.
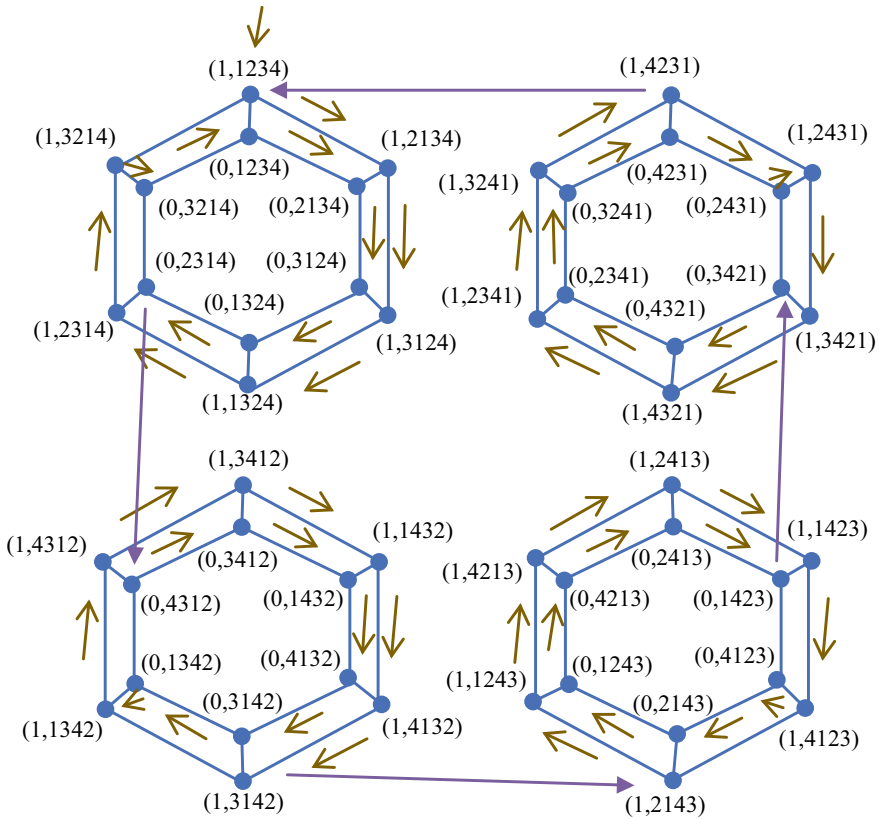
Fig. 5 Ring embedding in DS_4

To embed a ring in a DS_4 network, 48 edges are needed which can be written as $48 = (2 \times n!)$. All the nodes in DS_4 are connected using the star link and are described below to form a ring.

Cluster 1

$R_1 = (1.1234) \rightarrow R_2 = (1.2134) \rightarrow R_3 = (1.3124) \rightarrow R_4 = (1.1324) \rightarrow$
$R_5 = (1.2314) \rightarrow R_6 = (1.3214) \rightarrow R_7 = (0.3214) \rightarrow R_8 = (0.1234) \rightarrow$
$R_9 = (0.2134) \rightarrow R_{10} = (0.3124) \rightarrow R_{11} = (0.1324) \rightarrow R_{12} = (0.2314)$

Cluster 2

$R_{13} = (0.4312) \rightarrow R_{14} = (0.3412) \rightarrow R_{15} = (0.1432) \rightarrow R_{16} = (0.4132) \rightarrow$
$R_{17} = (0.3142) \rightarrow R_{18} = (0.1342) \rightarrow R_{19} = (1.1342) \rightarrow R_{20} = (1.4312) \rightarrow$
$R_{21} = (1.3412) \rightarrow R_{22} = (1.1432) \rightarrow R_{23} = (1.4132) \rightarrow R_{24} = (1.3142)$

Cluster 3

$R_{25} = (1.2143) \rightarrow R_{26} = (1.1243) \rightarrow R_{27} = (1.4213) \rightarrow R_{28} = (1.2413) \rightarrow$
$R_{29} = (1.1423) \rightarrow R_{30} = (1.4123) \rightarrow R_{31} = (0.4123) \rightarrow R_{32} = (0.2143) \rightarrow$
$R_{33} = (0.1243) \rightarrow R_{34} = (0.4213) \rightarrow R_{35} = (0.2413) \rightarrow R_{36} = (0.1423)$

Cluster 4

$R_{37} = (0.3421) \rightarrow R_{38} = (0.4321) \rightarrow R_{39} = (0.2341) \rightarrow R_{40} = (0.3241) \rightarrow$
$R_{41} = (0.4231) \rightarrow R_{42} = (0.2431) \rightarrow R_{43} = (1.2431) \rightarrow R_{44} = (1.3421) \rightarrow$
$R_{45} = (1.4321) \rightarrow R_{46} = (1.2341) \rightarrow R_{47} = (1.3241) \rightarrow R_{48} = (1.4231)$

Theorem 7 *Ring embedding is possible in DS_n, and the maximum size of the ring is $2n!$*

Proof In the double-star network, the total number of nodes is $2n!$ While constructing rings, it is always mandatory to touch all the nodes of the network. In Lemma 1, we get a total number of edges in a ring of DS_3 is $(2 \times 3!) = 12$.

Similarly, according to Lemma 2, the total number of edges in the DS_4 ring is $(2 \times 4!) = 48$. Hence, all the basic modules are connected in star fashion; hence, a ring can always be constructed in the double star of any dimension. So, we can say in a DS_n, the size of the ring will be $(2 \times n!)$ number of edges.

Here, the congestion of the embedding is one. Dilation and expansion are also one.

4.2   Mesh Embedding in DS

Lemma 3 *The maximum number of meshes possible in DS$_3$ is 6.*

In DS$_3$, the connection between the inner ring and outer ring for $n!$ number of nodes is $n!$ Hence, the maximum number of meshes in DS$_3$ is $n!$ which is 6. Meshes in DS$_3$ are shown in Fig. 6.

Lemma 4 *The maximum number of meshes possible in DS$_4$ is 36.*

For DS$_4$, there are four numbers of clusters consisting of DS$_3$. So, the maximum number of meshes in each cluster of DS$_4$ will be $(6 \times 4) = 24$. Then, all four clusters are connected in star fashion as shown in Fig. 7. The first cluster makes two sets of meshes with the remaining three clusters (mesh numbers 7, 29, 30, 31, 33, and 34).

Next, second cluster will make four numbers of meshes with the remaining two clusters (mesh numbers 14, 32, 35, and 36). Then, lastly, the third cluster will make two meshes using the cross-connections with the fourth cluster (mesh numbers 21 and 25).

So, these connections will embed a $(2 \times 3 + 2 \times 2 + 2 \times 1) = 12$ number of meshes. In total, there will be $24 + 12 = 36$ meshes embedded in DS$_4$.

Theorem 8 *The total number of meshes in DS$_n$ is $n! \times \left(\frac{n-1}{2}\right)$.*

Proof The DS network contains a DS$_3$ as the basic module, and it contains six numbers meshes. Then, the DS$_n$ contains n number of basic modules. To get a generalized equation for the calculation of the total number of meshes in DS$_n$, we can compare the above two mesh embedding in a double-star network. According to Lemmas 3 and 4, in DS$_3$, we get six meshes, and in DS$_4$, we get 36 meshes,

Fig. 6 Mesh embedding in DS$_3$

Fig. 7 Mesh embedding in DS$_4$

respectively. As the clusters are connected in star fashion, the number of edges connecting the clusters is given by $2[(n - 1) + (n - 2) + \cdots + 3 + 2 + 1]$.

Hence, the number of meshes possible in DS$_n$ is $6n + 2[(n - 1) + (n - 2) + \cdots + 3 + 2 + 1] = n! \times \frac{n-1}{2}$.

5 Conclusion

In the current research, the proposed topology is a new interconnection network that can be applicable for large-scale parallel systems. This DS network shows better characteristics than previous networks, such as diameter, cost, regular in nature, node degree, and traffic density, and also robustness. The value for degree and dimension of the DS graph is just like the hypercube. It contains the star structure. Mesh network can be easily embedded in the Double-star network is starting from dimension 3, whereas it was not at all possible in the case of star graph. The ring network is also proven to be embedded along with the Hamiltonian path. The dilation, congestion, and expansion of the embedding are unity. Thus, DS network is proved to be robust recursive and regular while retaining the original star topology.

References

1. Gram A. Parallel and distributed systems, 3rd edn. Tata McGraw Hill
2. Bhuyan LN, Agrawal DP (1989) Performance of multiprocessor interconnection network. IEEE Comput
3. Feng T (1981) A survey of interconnection networks. IEEE Comput 1(4):12–27
4. Akers SB, Krishnamurthy B (1987) The fault-tolerance of star graphs. In: Proceedings of international conference on supercomputing, p 270
5. Day K, Tripathy A. A comparative study of topological properties of hypercubes and star graphs. IEEE Trans Parallel Distrib Syst 5
6. Adhikari N, Nag B (2011) On topological properties of a star-based large scale parallel system. In: Proceedings of ETNCC2011, international conference on emerging trends in networks and computer communications, IEI Udaipur section, 22–24 Apr 2011
7. Adhikari N, Tripathy CR (2012) Mstar: a new two level interconnection network. In: Proceedings of 8th international conference on distributed computing and internet technology (ICDCIT'12), pp 50–61
8. Adhikari, N, Tripathy CR (2014) Starcrossed cube: an alternative to star graph. Turk J Electr Eng Comput Sci 22:719–734
9. Barik H, Swain N, Rout L, Adhikari N (2019) Double star: a high performance network for big data. In: SSRN eLibrary CTFC 2019
10. Swain N, Barik H, Adhikari N (2020) Load balancing in double star network. In: ICCSEA2020
11. Tripathy L, Tripathy CR (2021) Hierarchical hexagon: a new fault-tolerant interconnection network for parallel systems. Cybern Inf Technol 21(1):32–49
12. Abuelrub EM (1993) Interconnection networks embeddings and efficient parallel computations. LSU historical dissertations and theses. 5554. https://digitalcommons.lsu.edu/gradschool_diss theses/5554
13. Garey MR, Johnson DS (1979) Computers and intractability: a guide to the theory of NP-completeness. W. H. Freeman, New York, NY
14. Rahman MS, Kaykobad M, Firoz JS (2014) New sufficient conditions for Hamiltonian paths. Sci World J 2014:1–7
15. Mehedy L, Kamrul HM, Kaykobad M (2007) An improved degree based condition for Hamiltonian cycles. Inf Process Lett 102:108–112

A Comparative Study on Performance of Classification Algorithms for Breast Cancer Data Set Using WEKA Tool

Madhusmita Das and Rasmita Dash

Abstract Data mining is an important and contemporary tool for research. It comprises of different classification and prediction models. Data mining tools are used to summarize and extract useful information from different data bases. Most of the data base are consist of huge amount of unstructured data, and this unstructured data cannot be helpful to take a decision; therefore, different data mining tools are used to extract meaning full information from the datasets. Classification remains one of the widely used modern data mining tool for medical datasets. In this research work, various classification algorithms of data mining are implemented for breast cancer data set using Waikato Environment for Knowledge Analysis (WEKA) tool. The performance analysis of different classification algorithms can be further used to choose the best algorithm for the breast cancer data set analysis.

Keywords Data mining · WEKA tool · Breast cancer data set · Classification

1 Introduction

Databases are filled up with concealed information, and this concealed information can be used for intelligent decision making. Classification and prediction are two important tools for data analysis, and these tools can be used to design models. These models can describe important data classes and also can predict future data trends [1].

Now days, there are exponential increase in the implementation of different classification algorithms, so a single platform is required, where different algorithms can be implemented and compared their performance. WEKA is a standalone plat form,

M. Das (✉) · R. Dash
Institute of Technical Education and Research, Siksha 'O' Anusandhan, Deemed to be University, Bhubaneswar, India
e-mail: madhusmitadas@soa.ac.in

R. Dash
e-mail: rasmitadash@soa.ac.in

where different data mining technique such as classification, clustering, prediction, and many more can be implemented and their performance can be compared [1].

In this modern era of science, medical research is becoming an important subject, and data mining tools play a vital role in this research area. Medical data mining is an evergreen subject for the researcher, as medical data are pliantly available. Useful pattern can be discovered, from the medical dataset and different prediction can be made based on the data pattern. Among varies diseases, cancer is a chronic disease like diabetes, which needs extended treatment. Among different varieties of cancer, breast cancer is a life-threatening disease in females. In biomedicine, researchers try to calculate various outcomes. For example, a researcher will try to predict, if a growth found in a patient's breast is malignant or benign. To make this prediction, researchers use classification algorithms to compare test results and look for similarities. The algorithm provides a predictive model which can help to estimate whether or not the tumor is cancerous [2]. By looking at various statistics of the model such as model construction time, positively classified cases, kappa measure, True Positive Percentage (TPP), False Positive Percentage (FPP), and Receiver Operating Characteristic curve (ROC), researchers could determine whether the classifier made accurate predictions or not. Currently, there is a high probability of predicting breast cancer using data mining tools [2, 3]. In this research work, different classification algorithms are implemented in the WEKA tool for breast cancer dataset analysis.

The rest of the paper is organized as follows: in Sect. 2, a brief introduction to classifier is given. In Sect. 3, various methodology used and the data set used are discussed. Result study is done in Sect. 4 and to finish in Sect. 5 conclusion and future scope is set.

2 Classifier Used

In data mining, commonly used tools are the system, which construct classifier. The classifier takes a set of input cases in its training phase, and each case is fit to a definite class, describe by a fixed set of attributes. When a new case, with its associated attribute is input to the classifier, it can accurately predict its class. The performance of the classification algorithms varies. This variation is depended upon the type of the attribute, the type and size of data set [4].

A standard classification model can be designed after the through study of various previously available classification algorithm. The existing classification algorithms need to apply on different datasets and the performance and behavior of these existing algorithms should be consider for the design of a new algorithm. In this research work, total 64 classifier are used, on breast cancer data set using WEKA tool. These 64 classifiers are categorized to 6 different types of classifiers. They are as follows— Decision tree classifiers, Lazy learner classifiers, Meta classifiers, Misc classifiers, Rule classifiers, and Bayesian classifiers.

3 Methodology

3.1 Tool Used

The software tool used for this research work is WEKA 3.6.1, which is provided by the University of Waikato [3–5]. WEKA is the abbreviation of Waikato Environment for Knowledge Analysis. This tool is employed because it includes a variety of classification algorithms. WEKA is able to preprocess a dataset and then apply the selected classification algorithm using tenfold cross validation [6, 7]. During preprocessing, attributes can be removed from the dataset. The results of each test are formatted in a similar manner, which makes comparing the outcomes easier. The datasets accepted by WEKA are in ARFF format (Attribute Relation File Format) [8–10]. To provide and unbiased comparison, a significant variety of algorithms were used. This study compared 64 different classification algorithms that are available in WEKA. These algorithms were used because they can handle both numeric and nominal attributes. They also use different methods for building a predictive model.

3.2 Dataset Used

The dataset used in this research work is provided by the University of Wisconsin in the year 1992. The name of the data set is "UCI Breast Cancer Winconsin." In the used breast cancer dataset, total 699 instances are available. This dataset is be made up of 10 features or attributes along with its class. Here, the attribute value is of the type integer. Missing value is not present for any of the attribute in this data set. This dataset can be used to predict if a breast tumor is benign or malignant [2]. The data for this dataset is collected from fine needle aspirate test. This test uses a needle to collect cells from a suspicious lump. In this case, a suspicious lump is the breast cancer tumor.10-fold cross validation test mode is used in this research work.

4 Result and Discussion

Applying 8 different classifier of the type Byes classifier to the dataset breast cancer, the outputs are abridged in Table 1. As shown in Table 1, 8 different Bayesian classifiers are considered. From these 8 classifiers, naïve Bayes, naïve Bayes simple, and naïve Bayes updateable give the finest classification outcome. The ROC area of the classifiers also gives the same result that, naïve Bayes, naïve Bayes simple, and naïve Bayes updateable are best classifiers for breast cancer dataset.

Applying 5 different classifier of the type Lazy classifier to the dataset of breast cancer, the outcomes are abridged in Table 2.

Table 1 Comparison of various Bayesian classifiers over breast cancer data

Different classification models	Model construction time (measure in s)	Positively classified cases (measured in %)	Kappa measure	TPP	FPP	ROC
AODE	0	96.9253	0.9326	0.969	0.034	0.993
AODESr	0.2	96.7789	0.9296	0.968	0.033	0.989
Bayes net	0.28	97.3646	0.9425	0.974	0.024	0.993
HNB	0.02	96.0469	0.9122	0.96	0.058	0.99
Naïve Bayes	0.02	97.3646	0.9425	0.974	0.024	0.994
Naïve Bayes simple	0.02	97.3646	0.9425	0.974	0.024	0.994
Naïve Bayes updateable	0.02	97.3646	0.9425	0.974	0.024	0.994
WAODE	0.02	96.9253	0.9327	0.032	0.032	0.989

Table 2 Comparison of various lazy classifiers over breast cancer data

Different classification models	Model construction time (measure in s)	Positively classified cases (measured in %)	Kappa measure	TPP	FPP	ROC
IB1	0.02	95.1684	0.8929	0.952	0.067	0.943
IBK	0	95.9004	0.909	0.959	0.059	0.985
K-STAR	0	97.8448	0.9473	0.978	0.027	0.994
LBR	1.05	97.4138	0.937	0.974	0.029	0.993
LWL	0	91.3793	0.8027	0.914	0.053	0.977

As shown in Table 2, k-star is giving better accuracy by correctly classifying instance with 97.8448%. The ROC area of the k-star classifier is also more as compared to other classifiers.

Applying 10 different classifier of the Rules classifier to the breast cancer dataset, the results are abridged in Table 3. In Table 3, it is clearly visible, DTNB is the best classifier having highest accuracy for classification and the ROC area of DTNB is also maximum, which means among Rules classifier DTNB is the best classifier when it is apply to breast cancer dataset.

In the type, MISC classifier, two different variety of this classifier is applied to the dataset of breast cancer and their results are abridged in Table 4. Between the 2 classifiers of the category MISC, VFI is giving the highest accurateness and its ROC is also display that, so VFI is the better classifier in MISC category for breast cancer dataset.

Applying 14 different classifier of the type Tree classifier to the breast cancer dataset, the outputs are abridged in Table 5.

Table 3 Comparison of various rules classifiers over breast cancer data

Different classification models	Model construction time (measure in s)	Positively classified cases (measured in %)	Kappa measure	TPP	FPP	ROC
CONJUCTIVE RULES	0.05	86.6379	0.7101	0.866	0.062	0.902
DECISION TABLE	0.28	91.3793	0.7992	0.914	0.071	0.978
DTNB	0.94	96.5517	0.9168	0.966	0.032	0.992
JRIP	0.28	94.3966	0.8654	0.944	0.059	0.941
NNGE	0.25	96.1207	0.9077	0.961	0.025	0.968
ONE R	0.02	91.3793	0.7992	0.914	0.071	0.922
PART	0.08	92.2414	0.8209	0.922	0.049	0.981
PRISM	0.09	91.8103	0.9076	0.964	0.075	0.925
RIDOR	0.13	88.3621	0.699	0.884	0.22	0.832
ZERO R	0	71.5517	0	0.716	0.716	0.5

Table 4 Comparison of various MISC classifiers over breast cancer data

Different classification models	Model construction time (measure in s)	Positively classified cases (measured in %)	Kappa measure	TPP	FPP	ROC
HYPERPIPES	0.02	96.1207	0.9043	0.961	0.061	0.958
VFI	0.02	96.5517	0.9168	0.966	0.032	0.978

It is clear by Table 5, that NB tree is the best classifier among different Tree classifier. NB tree produce highest accuracy and its ROC is also maximum, so among tree, NB is the better classifier when it is applied to the dataset of breast cancer.

Applying 25 different classifier of the type Meta classifier to the breast cancer data set, the results are abridged in Table 6.

After analyzing Table 6, it is concluded that rotation forest classifier among different Meta classifier is the better one. The ROC of rotation forest, also gives best result, when it is applied to breast cancer data. In this research work, all together 64 classifiers are applied to the breast cancer dataset. These 64 classifiers are belonged to 6 different categories such as Bayesian, lazy, tree, rules, Meta, and Misc classifiers. The best classifier of each category is summarized in Table 7, with its classifier category.

Finally, all the 6, best classifier with its parameter such as model construction time, positively classified cases, kappa measure, True Positive Percentage (TPP), False Positive Percentage (FPP), and Receiver Operating Characteristic curve (ROC) are potted in Table 8. If correctly classified instances are considered as the parameter to choose the best classifier, then LBR is the best classifier among the 6 other classifiers

Table 5 Comparison of various TREE classifiers over breast cancer data

Different classification models	Model construction time (measure in s)	Positively classified cases (measured in %)	Kappa measure	TPP	FPP	ROC
AD TREE	0.05	95.3148	0.8964	0.953	0.062	0.989
BF TREE	1.2	95.4612	0.955	0.955	0.059	0.938
DECISION STUMP	0.03	87.5549	0.7435	0.876	0.082	0.881
FT	1.72	95.6076	0.9035	0.956	0.053	0.962
ID3	0.08	91.6545	0.9133	0.962	0.047	0.938
J48	0.03	93.4114	0.855	0.934	0.08	0.956
J48 GRAFT	0.11	93.5578	0.8573	0.936	0.085	0.954
LAD TREE	1.56	95.3148	0.8968	0.953	0.058	0.989
LMT	13.16	95.754	0.958	0.052	0.052	0.991
NB TREE	0.84	97.0717	0.936	0.971	0.029	0.99
RANDOM FOREST	0.19	96.3397	0.9196	0.963	0.043	0.988
RANDOM TREE	0	95.3148	0.8966	0.953	0.06	0.972
REP TREE	0.08	95.022	0.8902	0.95	0.064	0.973
SIMPLE CART	1.25	95.022	0.8912	0.95	0.054	0.945

Table 6 Comparison of various META classifiers over breast cancer data

Different classification models	Model construction time (measure in s)	Positively classified cases (measured in %)	Kappa measure	TPP	FPP	ROC
ADABOOST M1	0.13	95.4612	0.8996	0.955	0.061	0.989
ATTRIBUTE SELECTED CLASSIFIER	0.45	93.4114	0.855	0.934	0.08	0.956
BAGGING	0.16	95.754	0.9068	0.958	0.05	0.978
CLASSIFICATION VIA CLUSTERING	0.06	90.6296	0.7862	0.906	0.149	0.879
CLASSIFICATION VIA REGRESSION	1.56	96.0469	0.9129	0.96	0.05	0.987
CV PARAMETER SELECTION	0.02	65.0073	0	0.65	0.65	0.495
DAGGING	2.58	96.6325	0.9261	0.966	0.039	0.992
DECORATE	0.5	96.9253	0.9329	0.969	0.03	0.984

Table 6 (continued)

Different classification models	Model construction time (measure in s)	Positively classified cases (measured in %)	Kappa measure	TPP	FPP	ROC
END	0.02	93.414	0.855	0.934	0.08	0.956
ENSEMBLE SELECTION	4.88	95.6076	0.9033	0.956	0.055	0.976
FILTERED CLASSIFIER	0	93.4114	0.855	0.934	0.08	0.956
GRADING	0.03	65.0073	0	0.65	0.65	0.5
LOGITBOOST	0.09	96.0469	0.9132	0.96	0.046	0.992
MULTIBOOST	0.14	93.1186	0.8518	0.931	0.062	0.979
MULTICLASS CLASSIFIER	2.44	93.9971	0.8659	0.94	0.088	0.957
MULTISCHEME	0.02	65.0073	0	0.65	0.65	0.495
ORDINAL CLASS CLASSIFIER	0.02	93.4114	0.934	0.934	0.08	0.956
RACED INCREMENTAL LOGITBOOST	0	65.0073	0	0.65	0.65	0.495
RANDOM COMMITTEE	0.06	96.7789	0.9293	0.968	0.037	0.989
RANDOM SUBSPACE	0.14	95.6076	0.9035	0.956	0.053	0.984
ROTATION FOREST	4.52	97.3646	0.9426	0.974	0.022	0.992
STACKING	0.02	65.0073	0	0.65	0.65	0.495
STACKING C	0.02	65.0073	0	0.65	0.65	0.495
THRESHOLD SELECTOR	2.45	93.265	0	0.933	0.081	0.958
VOTE	0	65.0073	0	0.65	0.65	0.495

Table 7 Best classifiers over breast cancer dataset with its classifier category

Name of the classifier category	Best classifier
Tree classifier	NB tree classifier
Meta classifier	Rotation forest classifier
Misc classifier	VFT classifier
Rule classifier	DTNB classifier
Lazy classifier	LBR classifier
Bayesian classifier	Naïve Bayes classifier

Table 8 Comparison of the best classifiers of 6 different categories

Different classification models	Model construction time (measure in s)	Positively classified cases (measured in %)	Kappa measure	TPP	FPP	ROC
Naïve Bayes classifier	0.02	97.3646	0.9425	0.974	0.024	0.994
LBR classifier	1.05	97.4138	0.937	0.974	0.029	0.993
DTNB classifier	0.94	96.5517	0.9168	0.966	0.032	0.992
VFT classifier	0.02	96.5517	0.9168	0.966	0.032	0.978
Rotation forest classifier	4.52	97.3646	0.9426	0.974	0.022	0.992
NB tree classifier	0.84	97.0717	0.936	0.971	0.029	0.99

as mentioned in Table 8. However, if the ROC is considered as the parameter to select the best classifier, then naïve Bayes is the best classifier among all others.

5 Conclusion and Future Scope

The classification is an important data mining tool is a remarkable topic to the researcher. In this paper, different classification technique that is Bayes classifier, Lazy classifier, Meta classifier, Rules classifier, Misc classifier, and Tree classifier are consider in WEKA tool, to classify the breast cancer dataset. From the experiment, it is revealed that LBR classifier, which is belong to the category of LAZY classifier gives 97.4138% of correctly classified instance, when it is applied to breast cancer data set. However, ROC curve of the LBR classifier is not giving the best result, where as if we consider the ROC curve, as a parameter to determine the best classifier, then naïve Bayes classifier is giving better result among all the classifiers with 0.994 ROC curve area, which is the highest among all other classifier. There are too many different variables that affect the accuracy of a classification algorithm. One algorithm might perform well on one set of attributes, but if the attributes are modified then another algorithm might outperform it. Overall, no algorithm always performed the best. It depends on which attributes are being used. LBR and naïve Bayes classifiers gave the best results for the breast cancer dataset. In order to get the best results, it is good to use many different algorithms and then compare their predictive model. Trying different attribute sets can also lead to better predictive models.

References

1. Lavanya D (2012) Ensemble decision tree classifier for breast cancer data. Int J Inf Technol Conver Serv 2(1):17–24
2. Das M, Dash R (2020) Performance analysis of classification techniques for car data set analysis. In: 2020 international conference on communication and signal processing (ICCSP), pp 0549–0553
3. Anjana RM, Ali MK, Pradeepa R, Deepa M, Datta M, Unnikrishnan R, Rema M, Mohan V (2011) The need for obtaining accurate nationwide estimates of diabetes prevalence in India—rationale for a national study on diabetes. Indian J Med Res 133(4):369–380
4. Pandey UK, Pal S (2011) Data mining: a prediction of performer or underperformer using classification. Int J Comput Sci Inf Technol (IJCSIT) 2(2):686–690
5. Reyaz M, Kumar G (2018) Proficiency analysis of various data mining techniques for diabetes prognosis. Int J Comput Sci Eng 6(6):990–997
6. David SK, Saeb ATM, Al Rubeaan K (2013) Comparative analysis of data mining tools and classification techniques using WEKA in medical bioinformatics. Comput Eng Intell Syst 4(13):28–38
7. Jan S, Sharma V (2016) A study of various data mining techniques for diabetic prognosis. Int J Mod Comput Sci (IJMCS) 4(3)
8. Singh N, Singh D (2012) Performance evaluation of K-means and hierarchal clustering in terms of accuracy and running time. Int J Comput Sci Inf Technol (IJCSIT) 3(3):4119–4121
9. Mounika M, Suganya SD, Vijayashanthi B, Krishna Anand S (2015) Predictive analysis of diabetic treatment using classification algorithm. Int J Comput Sci Inf Technol (IJCSIT) 6(3):2502–2505
10. Suresh Kumar P, Umatejaswi V (2017) Diagnosing diabetes using data mining techniques. Int J Sci Res Publ 7(6)

An Optimum Segmentation of Gear Vibration Signals for an Effective Fault Classification Using Time-Domain Feature and Multi-class Support Vector Machines

Priyom Goswami, Prashant Kumar Sahu, and Rajiv Nandan Rai

Abstract Gear fault diagnosis has gained substantial attention over the last few decades as considerable amount of downtime for repairs and cost is associated with the transmission system including the gearbox. Three labelled data sets of gears are being analysed for fault classification, viz. no-fault, chipped tooth and worn tooth, for which three approaches are being explored to detect and classify fault. Two of the approaches are conventional using time-domain feature and time–frequency domain approach. The third approach being a hybrid procedure utilises time-domain statistical features along with multi-class support vector machine (SVM). For using SVM, the signal needs to be optimally segmented, and time-domain statistical features are then derived from each segment. These features obtained are then fed into a multi-class SVM model and are able to classify fault with 100% accuracy.

1 Introduction

The power transmission from one shaft to another for speed reduction or torque conversion can generally be obtained through a gearbox, chain sprocket or belt and pulley mechanisms. In contrast to the gearbox, belt–pulley and chain sprocket mechanisms are easy to operate and maintain and come with a cheap price tag compared to a gearbox. Belt–pulley and chain sprocket mechanisms are generally used wherever the power transmission load is low, the central distance between the shafts is high, and

P. Goswami (✉) · P. K. Sahu · R. N. Rai
Subir Chowdhury School of Quality and Reliability, IIT Kharagpur, Kharagpur, West Bengal 721302, India
e-mail: priyom.goswami@iitkgp.ac.in

P. K. Sahu
e-mail: prashantphd1@iitkgp.ac.in

R. N. Rai
e-mail: rnrai@iitkgp.ac.in

also in cases where no extreme difference in velocity ratio is required at the output shaft. The gearbox provides numerous advantages over the other two mechanisms, such as high torque transmission capability, less vibrations and noise during power transmission, maintaining constant velocity ratio at the output, compact construction and increased service life at the expense of cost associated with it. Gears are an indispensable part of any machinery, be it a huge production line machinery or the analogue watch in our hand.

In a survey done in 2005, it was found that around 74.7% of the time, gear fails due to user-related causes, and 25.3% of the time, gear fails due to design-related errors [1]. The design error includes errors during the manufacturing process, raw material defects and heat treatment errors. User-related cause for gear failure includes continuous overloading of gears, high level of speed and torque fluctuations, misalignment of gear shafts, impact loading, contaminated and improper use of lubrication, incorrect handling, operator error and so on.

Even though the gears are less prone to failures, the downtime and cost associated with failures are considerably high [2]. As per a study conducted on wind turbines, it was observed that gear faults count for only 9.8% of the total failures that lead to 19.4% of the total downtime [3]. Sometimes the failure of gears could even be life threatening. In 2009, an unexpected fault in the gearbox of a helicopter led to the loss of sixteen people's life [4]. Hence, gears can be termed as a critical component in most machinery, whose failure could incur loss of time, money and human life. So, it is of great importance to have a good condition monitoring system to reduce/prevent gear failures. Prognosis and health management (PHM) is emerging as a discipline for real-time assessment of gear conditions. Figure 1 represents the chronology of PHM tasks. Each task in PHM is interdependent.

The diagnosis process in the PHM provides the current health of the gears that includes fault detection, fault identification, fault classification, fault severity level analysis, etc. Prognosis deals with the remaining useful life (RUL) prediction of the gear based on the evidence produced by the diagnosis and historical data. Finally, the decision-making step includes the action to be implemented based on the observations made on the two previous steps. The decision-making step ensures that the gear life cycle cost is reduced, a safe operation can be maintained, and increased machine/system availability by expanding the maintenance cycle and reducing in-between inspection requirements, timely repair actions and improved logistic support [5].

Various techniques of monitoring fault in a machine are vibration monitoring, oil analysis, thermography, acoustic emission monitoring techniques, etc. Out of all the techniques, vibration monitoring is widely accepted due to its cost effectiveness and
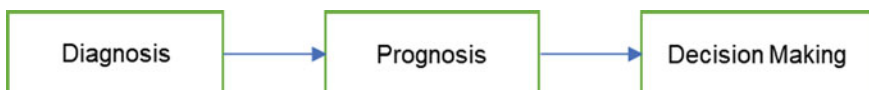

Fig. 1 Chronology of PHM tasks

its ability to detect almost all kinds of fault [6]. The data used for analysis in this paper is also based on the vibration monitoring technique. Time domain, frequency domain and time–frequency domain tools are being widely used in literature for gear fault diagnosis using vibration monitoring [7]. Generally, for fault classification and fault severity level identification and for finding RUL, machine learning (ML) tools like support vector machines (SVM), neural networks (NN) and random forest (RF) are widely used [8–10].

In this paper, a labelled data set of gear already existing in an online database [11] is used for the purpose of fault classification only. Here, two different faults are artificially induced on the gear, and signals are obtained for these faulty and healthy cases. The signals are then labelled for three different conditions of gears, viz. no-fault condition, chipped tooth and worn tooth of the gear. Details on induction of faults are provided in Sect. 2.1. Vibration monitoring techniques are data intensive; hence, it poses a challenge to classify defects with just three signals available and each signal having just one lakh (100,000) data points. In literature, fault classifications are generally based on a huge data set of gears [12] involving complex Gaussian correlation of vibrations signals and wavelet coefficients [13, 14]. These complex diagnosis and classification algorithm provide an accuracy ranging from 94 to 98% [13].

In this research paper, three approaches of fault classification have been considered, out of which, two are classical vibration monitoring fault classification approaches (time-domain and time–frequency domain approaches), and the third is a hybrid approach that comprises classical vibration monitoring and machine learning technique known as the multi-class SVM approach. As a result of the proposed approach, we are able to classify fault based on simple time-domain statistical features with 100% accuracy. As signal length also plays a vital role in classification, finding the optimal size for segmentation is also an important contribution of this research work. The paper is organised as follows: Sect. 2: Data and Experimental Set-up, Sect. 3: Methodology, Sect. 4: Results and Discussion and Sect. 5: Conclusion.

2 Data Set and Experiment Details

2.1 Experimental Set-up Details

The experimental set-up includes an electromotor, accelerometer, gearbox, brake system, analogue-to-digital (A/D) converter and a computer with MATLAB software installed for data processing. There are three different sets of gears for testing, viz. no-fault gear, chipped tooth and worn tooth. In the chipped tooth case, 50% of the area from top land to pitch surface is removed linearly with decreasing slope from a pinion gear tooth. However, for the worn case, 0.5 mm of material is removed from
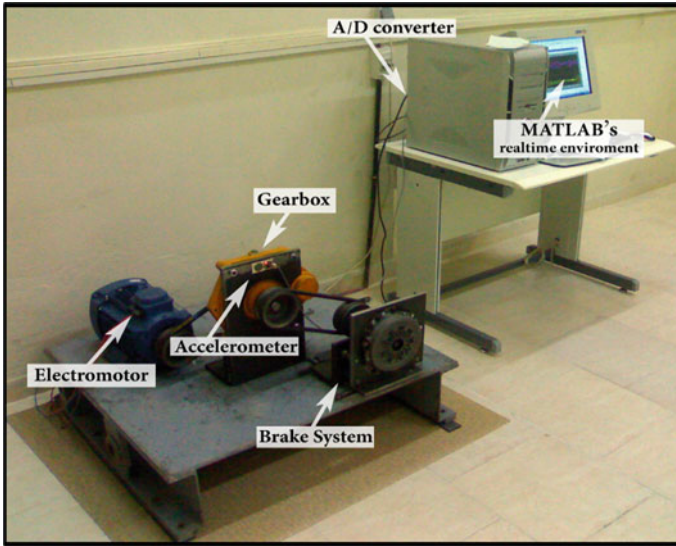
Fig. 2 Experimental set-up [13]
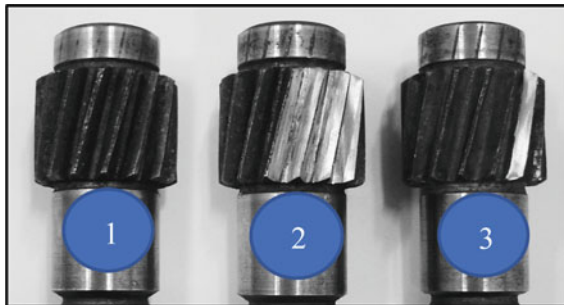
Fig. 3 Gears: (1) no-fault gear, (2) worn tooth and (3) chipped tooth [13]

the flank and face of three consecutive teeth [13]. The complete experimental set-up is shown in Fig. 2, and the healthy and faulty gears are shown in Fig. 3.

2.2 Data Set Details

The data set used in this paper for analysis is derived from an experiment conducted at the Amirkabir University of Technology. Three different labelled signals of no-fault condition, chipped tooth and worn tooth gear condition are used for the analysis. The complete data related to the experiment is provided in the tabulated form in Table 1.

Table 1 Specification of the experimental set-up

Experimental parameters	Corresponding values
Type of gears used	Helical
Pinion teeth count	15
Gear teeth count	110
Speed ratio	110/15 = 7.33
Input shaft speed	1420 rpm
Gear meshing frequency (GMF)	365 Hz (FFT analyser, actual)
Gear meshing frequency (GMF)	355 Hz (theoretical calculation)
Experiment run-time	10 s
Sampling frequency of accelerometer	10 kHz

3   Methodology

Three approaches of fault classification are taken into consideration for this paper. Out of the three approaches, two approaches are classical vibration monitoring fault classification approaches. The other is a hybrid approach that includes the use of classical vibration monitoring with an edge of the machine learning technique known as the multi-class SVM approach. Two classical approaches comprise (i) the time-domain approach and (ii) the time–frequency domain approach. The hybrid approach includes a combination of a machine learning model and a time-domain approach.

3.1   *Time-Domain Approach*

Raw vibration signals generated by an accelerometer from the gearbox comprise three different components, viz. (i) periodic component, due to the interaction between gear teeth meshing, (ii) transient component, due to short impact caused by tooth fault and (iii) broadband background noise [15]. The signal collected by the accelerometer over the gearbox housing may provide an erroneous signal at the output due to the interference of other signals from shafts, bearings, etc. The signal may also be contaminated with electrical resistance due to wired data transmission and may also be contaminated by electromagnetic interference.

   Time synchronous averaging (TSA) is one of the various ways through which frequency related to gears can be isolated from the raw signal that can contain noise due to other components like bearings, shafts, etc., of the gearbox. TSA helps separate rotational speed synchronous components from the other non-synchronous components, including the background broadband noise from the raw signal generated by

the accelerometer over the gearbox [16]. This aids in getting almost a noise-free gear signal from the raw cluttered signal [17].

The TSA is generally further processed for diagnosing several faults associated with gears, which cannot be interpreted simply from the raw TSA signal. The residual TSA is usually used for diagnosing gear faults like tooth cracks, chipped teeth, pitted tooth, etc. Residual TSA is generally obtained by removing the shaft rotational components, gear meshing frequency (GMF) and GMF harmonics from the original TSA signal. Hence, a residual signal only contains the sideband information of the gear signal. The increase in sideband amplitude can be used to diagnose or precisely classify the fault as chipped tooth fault [16].

For the case of worn teeth, the TSA is band passed around the GMF and GMF harmonics. Then, this band passed TSA signal is further processed such that the sidebands are removed across the GMF and its harmonics. The signal thus obtained is known as harmonic TSA. This harmonic TSA signal helps diagnose wear. Another way of classifying wear is by just applying fast Fourier transform (FFT) over the raw signal and observing for higher amplitude at higher harmonics of GMF. Generally, it is desired that a minimum of three GMF harmonics is considered for wear fault classification.

3.2 Time–Frequency Domain Approach

The spectral kurtosis (SK) is a frequency domain approach generally used to diagnose and classify fault in gears. SK is an established technique to capture small transients in the vibration signal [18]. Local faults like initial cracks and pitting mostly generate low-amplitude short duration transient in the raw vibration signal. These transients go undetected in the time-domain wavefront and also in the frequency spectrum obtained by applying FFT over the raw signal. It is because transients get submerged under the dominating component of the signal, as every section of the time-domain signal contributes to the FFT spectrum [18]. Hence, SK proves to be an efficient way for such a scenario.

Obtaining SK for a signal is not a straightforward path. Since these low-amplitude transients are not easily discernible, a kurtogram proves to be the need of the hour. Kurtogram provides the platform for applying spectrogram as kurtogram provides us with the information regarding bandwidth, central frequency and, most importantly, window length. The window length from the kurtogram is used to obtain the spectrogram. This spectrogram is then further used for SK analysis [19].

3.3 Multi-class SVM Approach

Support vector machine (SVM) is not a very new tool. Still, it is one of the most effective tools for classification problems due to its tendency of not overfitting even

Fig. 4 Graphical representation of three support vectors (two blue, one green) [20]
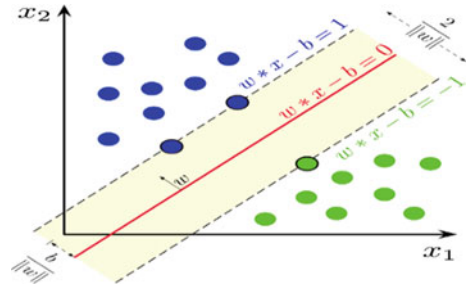
with low data availability. The sole objective of SVM is to find a hyperplane in an n-dimensional space that separates data points from the potential classes. The hyperplane is placed in such a way that it is at a maximum distance from the data points. The data points nearest to the hyperplane are known as support vectors, and they have the most decisive influence on forming the hyperplane at a given position. If the support vectors are removed, the plane also gets moved to a different location [20]. Figure 4 is a graphical representation of SVM, with tree support vectors.

The basic form of SVM does not support multi-class classification. For multi-class classification, the strategy of solving classification problems remains the same. The only modification is that the problem is broken down into smaller sub-problem, all of which are binary classification problems. There are many approaches to solve a multi-class classification problem, viz. one versus one, one versus all, directed acyclic graph, etc. Here in our case, we used one versus all approach. In one versus all approach, we train SVM equal to the number of classes for classification. To predict the output for new input, just predict with each of the build SVMs and then find which one puts the prediction the farthest into the positive region (behaves as a confidence criterion for a particular SVM) [21].

4 Results and Discussion

Analysing the raw signal in the time domain for fault classification has revealed interesting results. For the gear with chipped tooth fault classification, the raw vibration signal is time synchronous averaged and then the shaft frequency (i.e. 1420/60 = 23.667 rps), GMF (i.e. 365 Hz) and GMF harmonics (up to fifth-order harmonics) are removed to obtain the residual TSA signal in MATLAB. This residual signal is expected to have a higher amplitude of sideband around GMF. But such a scenario is not observed for our case, and for the worn-out teeth of gear, harmonic TSA has shown no significant effect on the amplitude of the higher GMF harmonics, as expected from the plot. A reason for such discrepancy can be noted because of

i.   The time synchronous averaging requires many revolutions to average out in order to produce good results. For this particular case of data set, the signal was recorded only for 10 s, which is significantly less and may prove to be a reason for not establishing a very good result [22].

ii. The signal might be masked heavily with background broadband noise.

The plots to justify the findings are shown in Figs. 5, 6 and 7.

Since using TSA as a time-domain approach failed to classify the faults in the gear signal, time-domain statistical features such as kurtosis, root mean square (RMS), maximum, minimum, skewness and crest factor are explored for each signal.

The statistical features of Table 2 are generally calculated from the signal to find a health indicator (HI) that can easily be used to classify the fault. But, from Table 2, no feature seems to be helpful in classifying fault. For differentiating faulty gear from non-faulty gear, kurtosis can only be used, as it can clearly be seen that kurtosis value is more than 3 for both the faulty cases. Apart from that, no other feature seems to be useful that can be used for classification visually. Kurtosis can only differentiate faulty from the non-faulty condition, which cannot be used to identify or classify the actual fault. Hence, only just using a time-domain statistical approach over the whole signal does not work very well for classification for this particular kind of data set.

So, the next time–frequency approach of spectral kurtosis (SK) is taken into consideration. Spectral kurtosis is a proven way to find the transient from a noisy signal. But for effective use of SK, the appropriate window length needs to be selected, as SK is greatly affected by window length. For this reason, a kurtogram needs to be plotted. The kurtogram displays the different values of SK calculated by using series of filter banks with various central frequency and bandwidth. Using the details obtained through kurtogram, the same is being used to find the short-time Fourier
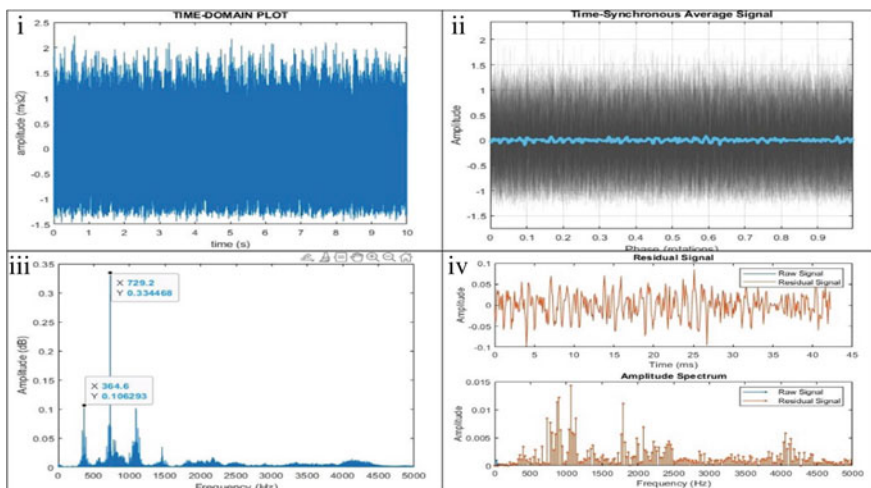
Fig. 5 Plots for no-fault condition (i) time-domain plot, (ii) TSA plot, (iii) FFT plot and (iv) residual TSA plot
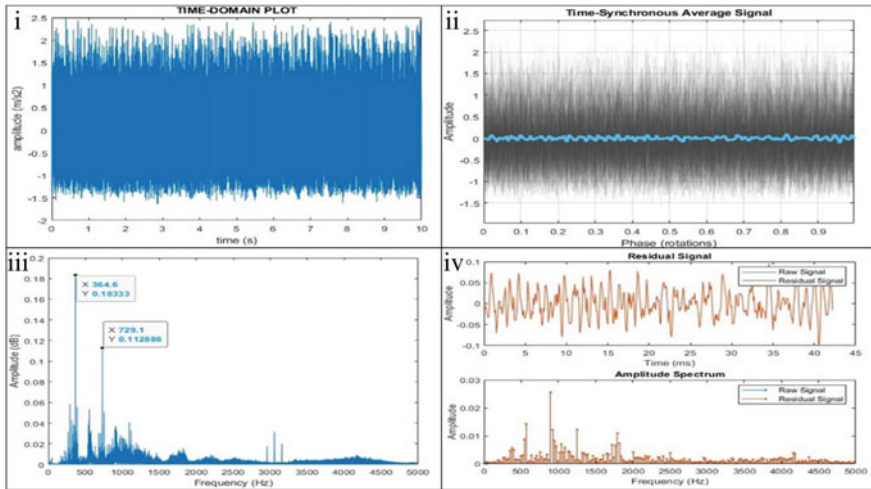
Fig. 6 Plots for chipped tooth condition (i) time-domain plot, (ii) TSA plot, (iii) FFT plot and (iv) residual TSA plot
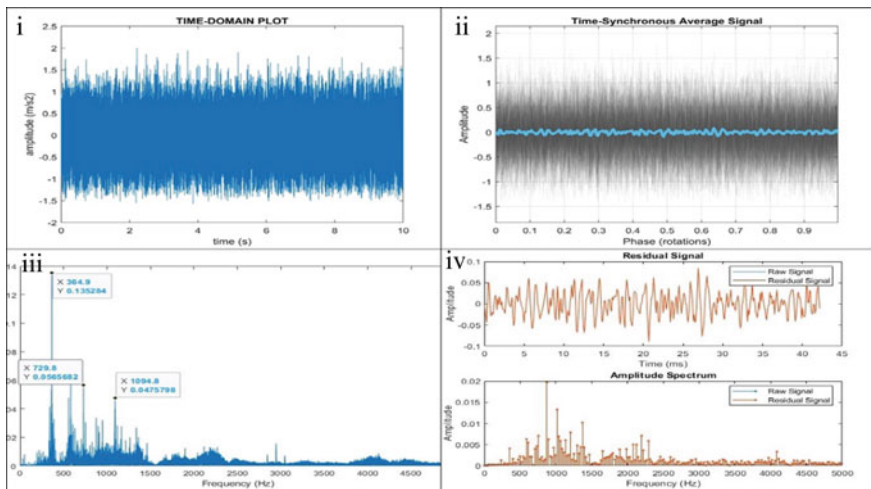
Fig. 7 Plots for worn tooth condition (i) time-domain plot, (ii) TSA plot, (iii) FFT plot and (iv) residual TSA plot

Table 2 Time-domain statistical features for gear signals

Condition/features	Kurtosis	RMS	Maximum	Minimum	Skewness	Crest factor
No fault	2.47	0.63	2.22	−1.48	0.14	3.53
Chipped tooth	3.27	0.62	2.48	−1.64	0.36	3.98
Worn tooth	3.33	0.47	2.00	−1.57	−0.03	4.21

transform (STFT) coefficients (t, f) of each time window. After that, SK can be calculated from the signal with all parameters known: the central frequency, the bandwidth, the window length and the STFT coefficients. For calculating STFT coefficients, the hamming window function is selected by default, and no overlap between windows is considered [19]. The results obtained are shown in Fig. 8.

Analysing all the SK plots (Fig. 8) obtained from three different gear conditions, there is no apparent reference through which classification can be made. Even though low-amplitude transients can be seen in the plot, the absence of a high value of spectral kurtosis that may be used as a classifier for fault classification is missing. This leads to failure in the classification of gear fault again. The two conventional

Fig. 8 Kurtogram and SK of gear for (i) no-fault condition, (ii) chipped tooth condition and (iii) worn tooth condition

fault classification approaches seem to be ineffective against the given data set for fault classification. The reason could be due to the low availability of data. Using only a 10 kHz accelerometer and running the experiment for just 10 s make fault classification difficult for conventional methods to classify fault. As all the processes mentio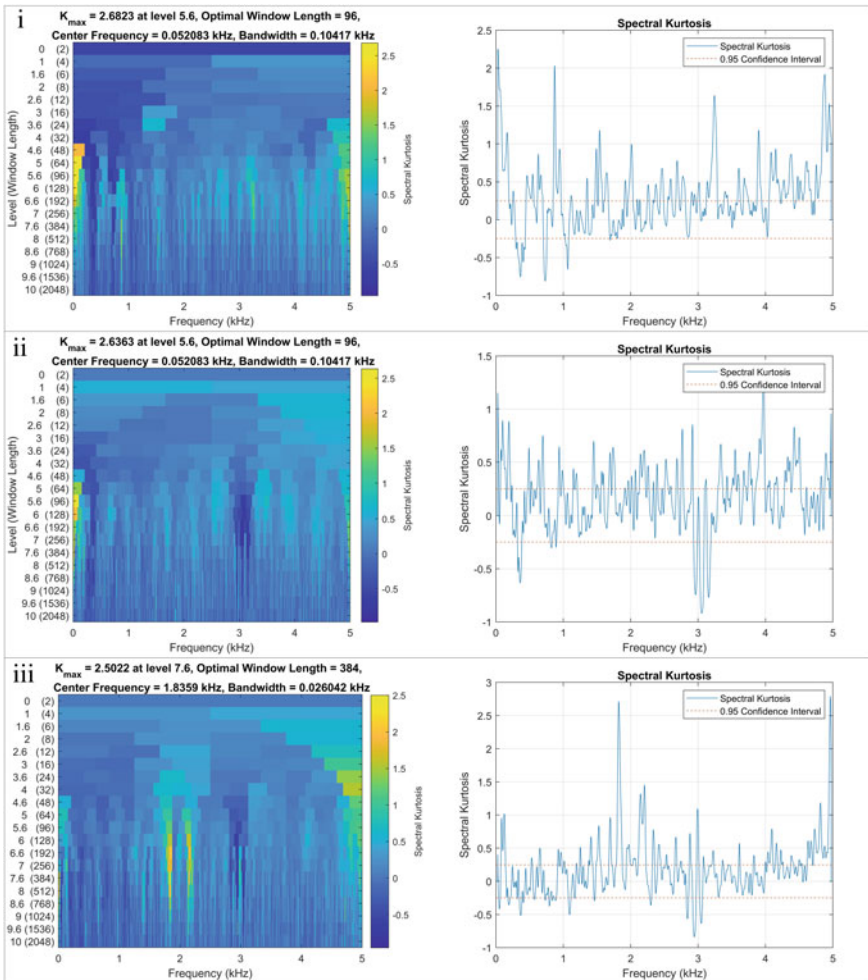ned above are data-driven approach, it requires a lot of data for processing. But in our case, we only have three signals with two different classes of fault and one without any fault. This less availability of data leads to inefficient classification mechanisms. Hence, a third hybrid approach was used for fault classification. The use of time-domain statistical features along with the multi-class support vector machines (SVM) approach is then used for gear fault classification.

It is pretty apparent that SVM can be used for multi-class classification problems with a smaller number of data points for training and testing; hence, this algorithm is selected. To use this SVM, the gears signals are divided into three classes—class 0: no-fault, class 1: chipped tooth and class 2: worn tooth. The vibration signal of each class has been captured for 10 s with a sampling frequency of 10 kHz. Thus, each signal contains 100,000 data points. Now to generate data for the SVM algorithm, the signals are divided into several small segments. It is necessary to select the optimum number of data points for effective analysis. By selecting fewer data points, signal aliasing will occur, which may lead to loss of fault information. However, choosing a larger number of data points for each segment leads to a lesser number of features and will not be sufficient for testing and training set for ML algorithms.

In this work, two signals' segments are selected for analysis: one with 1000 data points and the other with 2000 data points. The optimum signal length is chosen by comparing the time-domain kurtosis features of the segmented signal with the original signal, as shown in Fig. 9.

The kurtosis value for a segmented signal containing 1000 data points shows a considerable deviation from the original signal in the case of a chipped tooth and
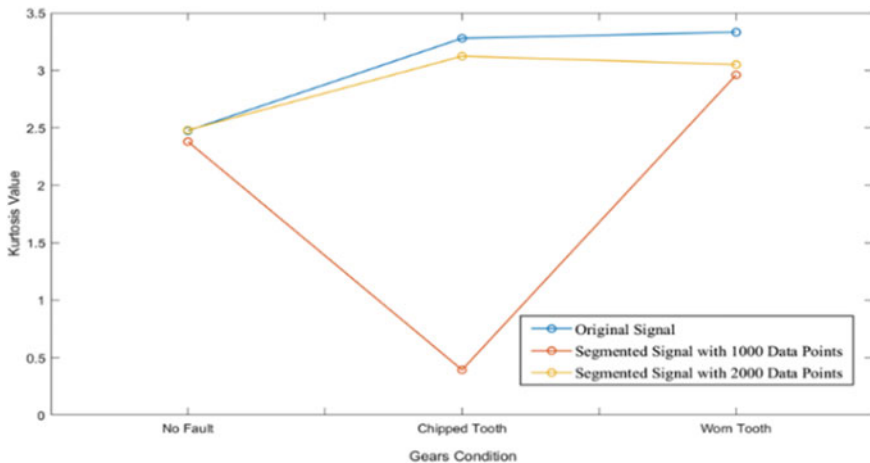

Fig. 9 Kurtosis comparison between original and segmented signal
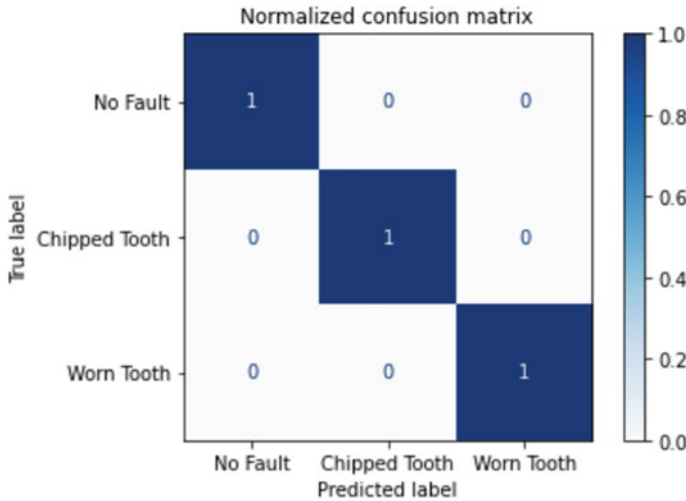
Fig. 10 Normalised confusion matrix using SVM for gears fault classification

loses the fault characteristics of the signal, which is mainly due to under-sampling. Thus, more data is required for signal segmentation. However, the kurtosis value for a segmented signal containing 2000 data points, i.e. 50 features, is closely related to the original signal and considered for further analysis.

The time-domain statistical features such as kurtosis, root mean square (RMS), maximum, minimum, skewness and crest factor for a segmented signal with 2000 data points are selected as input parameters for the kernel SVM using the radial basis function (RBF). The combined features are found to be effective health indicators for the gears fault classification, which shows a classification accuracy of 100%, as shown in the confusion matrix in Fig. 10.

5 Conclusion

This paper dealt with three approaches of fault classification: two classical and one hybrid. The results obtained through these three processes were then analysed for fault classification. The two classical approaches failed to classify the incipient fault for a situation where the data points available from the signal are quantitatively less. In our case, we only have three signals with data just recorded for 10 s and a cumulative of 1 lakh data points for each signal. Hence, for proper classification of fault, we need a machine learning algorithm that requires less data for training as well as testing. For our case, the multi-class SVM performed exceptionally well even with very low available data. The complete objective of fault classification is therefore achieved by segmenting the signal into optimum length of 2000 data points per signal and then extracting time-domain features from each of these 50 segments. After analysing

these time-domain features using multi-class SVM, a 100% accurate classification of fault was achieved. Hence, machine learning plays a vital role in the field of PHM, be it fault classification or RUL prediction. This domain of PHM would not have evolved to the stage now without machine learning and artificial intelligence.

References

1. Davis JR (2005) Gear materials, properties & manufacture. ASM International
2. Astridge DG (1989) Helicopter transmission design for safety and reliability. J Aerosp Eng 203:123–138
3. Ribrant J, Bertling LM (2007) Survey of failures in wind power system with focus on Swedish wind power plants during 1997–2005. IEEE Trans Energy Conserv 22:167–173
4. Department for Transport Air Accidents Investigation (2011) Aircraft accident report 2/2011. [Online]. Available: https://assets.publishing.service.gov.uk/media/5422f86aed915d13710000 6cb/2-2011_G-REDL.pdf. Accessed 22 Aug 2021
5. Pecht M, Kang M. Prognosis and health management of electronics: fundamentals, machine learning and the internet of things. Wiley, Chichester
6. Yang W, Tavner PJ, Wilkinson MR (2009) Condition monitoring and fault diagnosis of a wind turbine synchronous generator drive train. IET Renew Power Gener 3(1):1–11
7. Sait AS, Sharaf-Eldeen YI (2011) A review of gearbox condition monitoring based on vibration analysis techniques diagnosis and prognosis. In: Conference proceedings of the society for experimental mechanics series
8. Qu J, Liu Z, Zuo MJ (2011) Feature selection for damage degree classification of planetary gearboxes using support vector machine. J Mech Eng Sci 225:2250–2264
9. Saravanan N, Ramachandran KI (2010) Incipient gearbox fault diagnosis using discrete wavelet transform (DWT) for feature extraction and classification using artificial neural network (ANN). Expert Syst Appl 37:4168–4181
10. Cerrada M, Zurita G, Cabrera D (2016) Fault diagnosis in spur gears based on genetic algorithm and random forest. Mech Syst Signal Process 70:87–103
11. Zamanian AH. Experimental dataset for gear fault diagnosis. [Online]. Available: https://goo. gl/TorZJq. Accessed 22 July 2021
12. Yang J, Xie G, Yang Y, Zhang Y, Liu W (2019) Deep model integrated with data correlation analysis for multiple intermittent faults diagnosis. ISA Trans 95:306–319
13. Zamanian AH, Ohadi A (2011) Gear fault diagnosis based on Gaussian correlation of vibrations signals and wavelet coefficients. Appl Soft Comput 11:4807–4819
14. Zamanian AH, Ohadi A (2010) Gearbox fault detection through PSO exact wavelet analysis and SVM classifier. In: 18th annual international conference on mechanical engineering-ISME2010, Tehran, Iran
15. Al-Arbi S. Condition monitoring of gear systems using vibration analysis. University of Huddersfield. [Online]. Available: http://eprints.hud.ac.uk/17821/. Accessed 06 Sept 2021
16. Kundu P, Darpe AK, Kulkarni MS (2019) An ensemble decision tree methodology for remaining useful life prediction of spur gears under natural pitting progression. Struct Health Monit 19:854–872
17. Fan Q, Zhou Q, Wu C (2017) Gear tooth surface damage diagnosis based on analyzing the vibration signal of an individual gear tooth. Adv Mech Eng 9:1–14
18. Kundu P, Darpe AK, Kulkarni MS (2020) A review on diagnostic and prognostic approach for gears. Struct Health Monit 1–41
19. Saidi L, Ali JB, Bechhoefer E, Benbouzid M (2017) Wind turbine high speed shaft bearings health prognosis through a spectral kurtosis derived indices and SVR. Appl Acoust 120:1–8

20. Marius H (2020) Towards data science: multiclass classification with support vector machines (SVM). [Online]. Available: https://towardsdatascience.com/multiclass-classification-with-support-vector-machines-svm-kernel-trick-kernel-functions-f9d5377d6f02. Accessed 27 Sept 2021
21. Goyal C (2021) Analytics Vidhya: multiclass classification using SVM. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/05/multiclass-classification-using-svm/. Accessed 28 Sept 2021
22. Barszcz T, Randall RB (2009) Application of spectral kurtosis for detection of a tooth crack in the planetary gear of a wind turbine. Mech Syst Signal Process 23:1352–1365

PhyMCons: Physical Machine Consolidation for Minimizing VM Migration and Energy Consumption in Fog Center

Lalbihari Barik, Mahmoud Al Ahmad, Mubashshirahbanu Shekh, Akash Barik, Sudhanshu Shekhar Patra, and Sibananda Behera

Abstract Virtualization technologies have revolutionized how businesses use their server resources in their fog centers. Rather than using dedicated physical machines for individual applications, virtualization makes it possible to view resources as a unified resource pool, simplifying management, and lowering complexity. One of the most common applications of virtualization in fog centers is physical machine consolidation, which involves lowering the number of physical machines used by consolidating applications. The latter technique aids in the more efficient use of computing resources and has numerous advantages, including lower power and cooling costs, and thus imparts to the Green IT program. In a data center environment that is always evolving, where virtual machines (VMs), lowering the number of physical machines utilized can be accomplished by migrating applications in a non-disruptive manner, a technique termed live migration. But since live migration is an expensive procedure, it's challenging to figure out how to execute periodic physical machine consolidation in a migration-aware manner. We present the PhyMCons physical machine consolidation algorithm, which reduces both the overall number of physical machines in operation, as well as the number of migrations which in turn

L. Barik
Department of Information Systems, Faculty of Computing and Information Technology, King Abdulaziz University, Rabigh, Kingdom of Saudi Arabia

M. A. Ahmad
School of Computer Engineering, KIIT Deemed to be University, Bhubaneswar, India

M. Shekh
Department of Computer Science, Shrimad Rajchandra Institute of Management and Computer Application, Uka Tarsadia University, Gujarat, India

A. Barik
Department of Computer Science, New Horizon College of Engineering, Bengaluru, India

S. S. Patra (✉)
School of Computer Applications, KIIT Deemed to be University, Bhubaneswar, India
e-mail: sudhanshupatra@gmail.com

S. Behera
Department of MCA, Trident Academy of Creative Technology, Bhubaneswar, India

reduces the energy consumption. We demonstrate the practicality of our approach as well as its scalability.

Keywords Virtualization · VM migration · Energy saving · Physical machine consolidation · Fog center

1 Introduction

The term "green IT" refers to IT solutions that are energy efficient. In fog computing environments, energy efficiency is critical. Service providers deliver their services through data centers, which contain thousands of processing nodes and require a lot of electricity. In fog centers, there are several options for lowering power consumption. Berl et al. [1] propose four methods for improving energy efficiency. Our research is focused on the third option of their defined options which involves condensing applications onto fewer servers in a virtualized cluster environment.

In fog centers, virtualization gives a unified picture of a pool of resources, allowing a single server to support many computing environments, each encased as a virtual machine and segregated from the others. One of the most common uses of virtualization technology in fog centers is physical machine consolidation. The first sort of server consolidation involves moving many applications from physical machines to a smaller number of high-performance virtualized physical computers. It is batch task that occurs infrequently. The second method of physical machine consolidation involves changing VM mappings by dynamically reallocating VMs while they are still running. The second sort of physical machine consolidation will be our focus in this research. Figure 1 shows the physical machine consolidation operation begins in an intermediate stage and proceeds via many VM migrations to arrive at a state where only the bare minimum of physical machines is used. We present PhyMCons, a physical machine consolidation approach that reduces the number of machines as well as the migrations number.

The PhyMCons method shares several characteristics with the well-known heuristic bin-packing algorithms FFD and BFD. However, while FFD and BFD are solely concerned with reducing the number of physical servers, PhyMCons is also concerned with reducing the number of migrations. The PhyMCons algorithm's
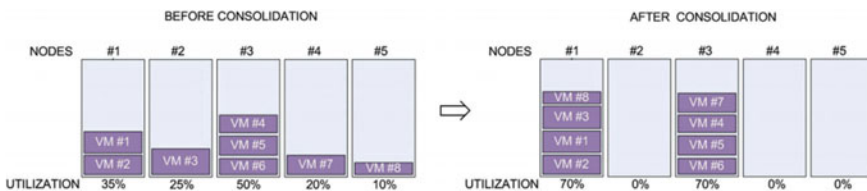

Fig. 1 Physical machine consolidation

main idea is in moving the VMs from the servers having less loaded to the heavily loaded ones while adhering to a set of constraints specified in the issue modeling, allowing the least loaded servers to be released.

The paper is structured as follows. Section 1 represents the introduction; Sect. 2 the related work; Sect. 3, Physical Machine Consolidation Modeling; Sect. 4, PhyMcons Algorithm; Sect. 5, the Energy Model; Sect. 6, an Illustration; Sect. 7 analyzed the result and at last Sect. 8 depicts the conclusion of the article.

2   Related Work

The live migration [2] allows users to shift a VM from one PM to another while preserving client service levels. The memory contents of a virtual machine can be transported from one physical host to another in a variety of methods. Various research papers [3–5] have looked into energy optimization during the process of VM Live Migration. The majority of existing or proposed studies look at VM's energy consumption under various circumstances, live migration in terms of migration time and downtime. It's hard to find research that examines the energy usage of VM live migration in certain subsystem restrictions. The physical machine consolidation operation is carried out by transferring virtual machines (VMs) employing the Live Migration technology from one node to another [6, 7], which is available in most common hypervisors like Xen and VMware. Live Migration, on the other hand, is an expensive procedure; some CPU processing on the migrating node is included in the costs of VM migration, connection bandwidth between the migrating and migrated nodes, service unavailability on the moving VM, and overall migration time [8, 9]. General packing challenges are connected to the issue of VM allocation across a cluster of nodes. One of these challenges is the bin-packing problem [10], in which a limited number of bins must be filled with a variety of sizes in order to employ the smallest bins. We can think of VM as an item and nodes as bins in our scenario. Although the bin-packing problem is NP-hard, there are a number of heuristic algorithms that can give sub-optimal results, such as Next-Fit (NF), First-Fit (FF), and Best-Fit (BF), all of which can be calculated instantly (O(n) for Next-Fit and O(n logn) for First-Fit and Best-Fit) [11].

Li et al. [12] had suggested an adaptive method which specified a few critical performance indicators for live migration. It tested the impact of many parameters on VM live migration on KVM, such as speed limit, maximum downtime, TCP buffer size, VM size, workload, and network bandwidth, as well as the energy usage. Strunk et al. [13] explored whether live migration results in energy overhead, which varies depending on the VM's size and the usable network bandwidth.

3   Physical Machine Consolidation Modeling

Live migration is an expensive procedure. On the migrating node, some CPU processing is required. Besides that the connection bandwidth between the migrating and migrated nodes, the service unavailability on the moving VM, and the total migration duration are essential. As a result, our primary goal is in reducing the number of migrations. The other goal is in consolidating the VMs to use the fewest number of nodes possible.

A CPU threshold must be set on an individual node. This is done to keep the CPU of a node below 100% utilization. The reason for this is 100% CPU utilization can result in performance loss. Additionally, the live migration technique uses some CPU cycles. Maintaining a specific level of CPU throughput requires keeping CPU use below that threshold number. Choosing an appropriate CPU threshold value is critical, as selection of high threshold reduces the performance of VMs operating on an individual node, where as a low threshold reduces the efficacy of consolidation. The appropriate CPU threshold value for consolidation has yet to be determined by the researchers. According to several articles, this number fluctuates from 50 to 75%. As a result, we chose values from this range for our experiment.

To describe a VM and a node in our work, we use two dimensions: CPU and memory.

Let a physical machine i be depicted as a vector P_i having two dimensions $Pi(c_i, m_i)$ where c_i the total CPU capacity in GHz and m_i the memory capacity measured in megabytes (MB). In a similar fashion, VM is denoted with the vector V_j (vc_j, vm_j), where vc_j is the CPU and vm_j is the memory capacity of the jth VM, respectively.

Let in the fog center there are M physical machines and K VMs. Let $<S_1, S_2, S_3, \ldots S_M>$ be set representing the PM's, and the PM S_i has the VM identification numbers as $\{i_1, i_2, \ldots i_p\}$.

The value of the CPU threshold is defined as T where T ranges between [0, 1]. It can be formalized as follows:

$$\frac{\sum_{j \in S_i} vc_j}{c_i} \leq T$$

$\forall$ PM i.

A set of VMs in the PM cannot have a total capacity greater than the PM's capacity.

There are 3 matrices that can calculate the efficacy of the physical machine consolidation algorithm.

1.   The # of PMs that were used
2.   The # of PM's that were released
3.   The # of VM's that were migrated

4 PhyMCons Algorithm

The proposed algorithm inherits some First-Fit and Best-Fit features, which aim to reduce the number of PMs utilized. Our proposed technique, on the other hand, not only reduces the required PMs, but also the migrations number.

As a result, this algorithm behaves differently than bin-packing methods. The method sorts the PMs based on the load carried by VMs, in non-increasing order. The last PM in the list which is the least loaded is chosen as a candidate for migration, and the weights of the VMs in that PM are arranged in decreasing order. We attempt to assign them one by one the first (most heavily loaded) PM, and if that fails, we move on to the second PM, and so on. The motivation for this approach is that at the start of the process, the PMs are sparsely loaded, so we aim to condense them by transferring VMs from the lowest loaded PMs to the highly loaded PMs so that the least burdened PMs can be released.

The stages are continued until there are no more migration options.

The PM by VMs is represented by two numbers: the VM's CPU and memory capacity. Let's call the CPU and memory loads on a PM i cl_i and ml_i, respectively, and derive them as follows:

$$cl_i = \frac{\sum_{j \in S_i} vc_j}{c_i}$$

$$ml_i = \frac{\sum_{j \in S_i} vm_j}{m_i}$$

The PMs can be sorted with a single value that can be used to represent the score of the machine. The score of a PM can be calculated as:

$$\text{score}(s_i) = \lambda cl_i + (1 - \lambda)ml_i$$

where λ can be calculated as follows:

$$\lambda = \frac{\sum_{i \in M} cl_i}{\sum_{i \in M} cl_i + ml_i}$$

Similarly, the scores of the Q VMs based on their CPU and memory parameters, vc_i and vm_i are:

$$\text{score}(v_i) = \lambda_v.cl_i + (1 - \lambda_v)vm_i$$

where

$$\lambda_v = \frac{\sum_{i \in Q} vc_i}{\sum_{i \in Q} (vc_i + vm_i)}$$

5   Energy Model

The energy model used is

$$\text{Pow}(U) = \begin{cases} \text{Pow}_{\text{min}} + (\text{Pow}_{\text{max}} - \text{Pow}_{\text{min}})X \ \ U & \text{if} \ \ U > 0 \\ 0 \ \ \text{otherwise} \end{cases}$$

where Pow_{min} is an idle server's power consumption, Pow_{max} is the power consumption when the CPU utilization is 100% and U is the current CPU utilization.

6   Illustration

Let the system has $M = 6$ nodes $S = <s_1, s_2, s_3, s_4, s_5, s_6>$ and $K = 15$ VMs $<v_1, v_2, v_3, \ldots v_{15}>$ with capacities of PM's in terms of CPU and Memory are {6 GHz, 4 GB} each. The capacity of the VMs from v_1 to v_{15} are {1, 0.5}, {1, 0.25}, {1.5, 1}, {1, 0.25}, {1.2, 0.5}, {1, 1}, {0.5, 0.5}, {0.5, 0.5}, {1, 1}, {1, 0.5}, {1, 0.75}, {1.5, 1}, {{1, 0.75}, {1, 05}, {0.5, 0.25}, respectively.

We have taken the CPU utilization threshold T as 0.7 or 70%. The initial stage of the VMs along with the PMs is shown in Fig. 2 and the score of each machine is defined.

Initial State

score(s1)   =   (0.546)(0.546)   +   (1–0.546)(0.4375)   =   0.51.score(s1)   =
(0.546)(0.546) + (1–0.546)(0.4375) = 0.51.
score(s2) = (0.546)(0.533) + (1–0.546)(0.4375) = 0.49.
score(s3) = (0.546)(0.083) + (1–0.546)(0.125) = 0.10.
score(s4) = (0.546)(0.25) + (1–0.546)(0.375) = 0.31.
score(s5) = (0.546)(0.333) + (1–0.546)(0.375) = 0.32.
score(s6) = (0.546)(0.667) + (1–0.546)(0.625) = 0.65.
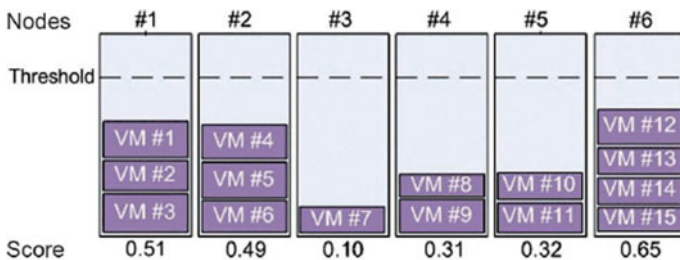
Fig. 2 Initial stages of the VMs with scores

The PMs should be ordered in decreasing order based on their score: <*s*6, *s*1, *s*2, *s*5, *s*4, *s*3>.

First Iteration: VM #7 migrated from #3 to #1 which is shown in Fig. 3.

Second Iteration: VM #9 migrated from #4 to #2, VM #8 migrated from #4 to #5 shown in Fig. 4.

Final Stage: The final stage of the consolidation is shown in Fig. 5.
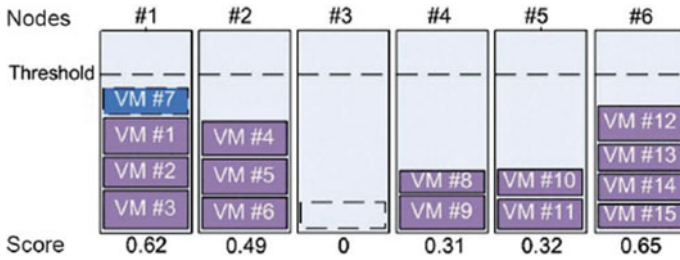
Fig. 3 The scores of the VMs after iteration 1
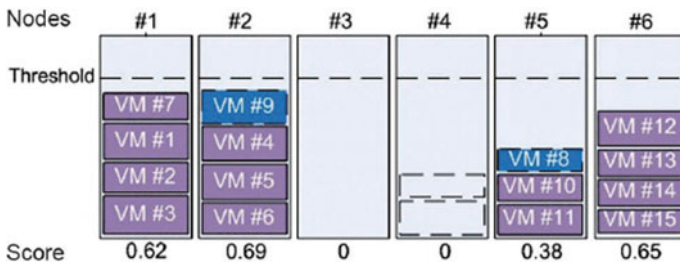
Fig. 4 The scores of the VMs after the second iteration
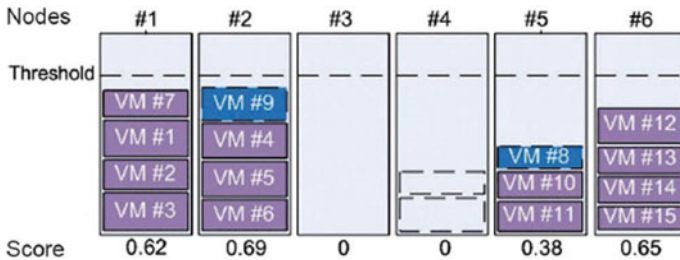
Fig. 5 The final stage of the VMs where VMs #3 and #4 can be switched off

7 Results Analysis

Our proposed algorithm is implemented under Matlab R2014a on an Intel(R) Core(TM) *i*5-8250*U* CPU @1.60 GHz 1.80 GHz CPU running on Windows 10 64-bit professional and 8 GB RAM.

Figure 6 shows the relation of # of migrations Vs # of VMs used. The figure shows as the # of VMs increases the # of migrations. In our proposed algorithm PhyMCons, the number of VMs migrated is less as compared to the other state of art algorithms for the same number of VMs. Figure 7 shows the number of VM used Vs the nodes used. The nodes used increases as # of VMs increases. PhyMCons gives better performance as compared to the other algorithms FFD, BFD and MBFD.

Figure 8 depicts the relationship between # of nodes released Vs # of nodes migrated. The number of VMs migrated increases as the number of nodes released increases. The PhyMCons have better performance as compared to the other state of art algorithms such as FFD, MBFD, and BFD. Figure 9 shows the relation between # of VMs used and the energy consumption where the energy consumption with our

Fig. 6 Relationship between # of migrations and # of VMs used
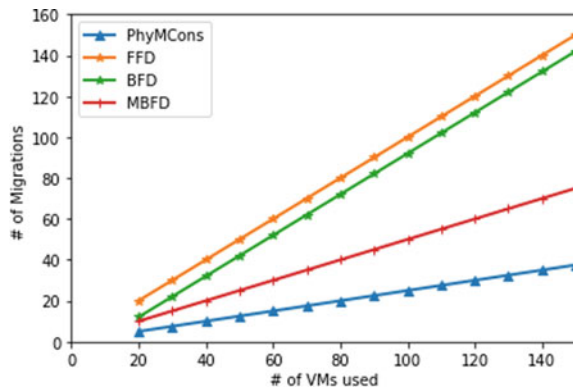
Fig. 7 Relationship between # of used nodes and # of used VMs
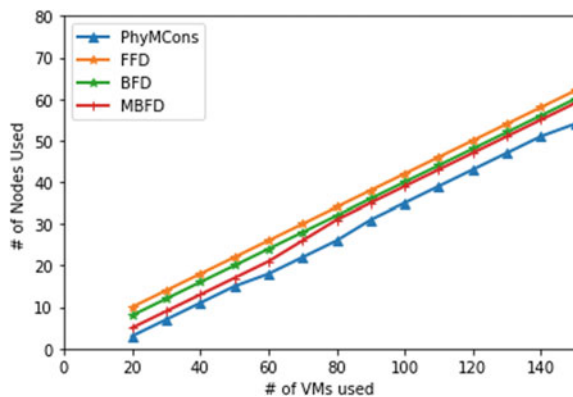
Fig. 8 Relationship between # of migrations and # of released nodes
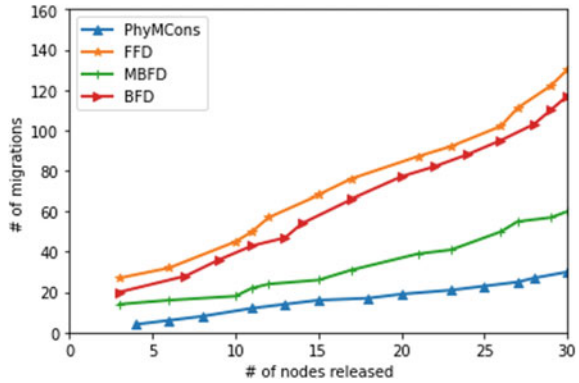
Fig. 9 Relationship between # of used VMs and energy consumption
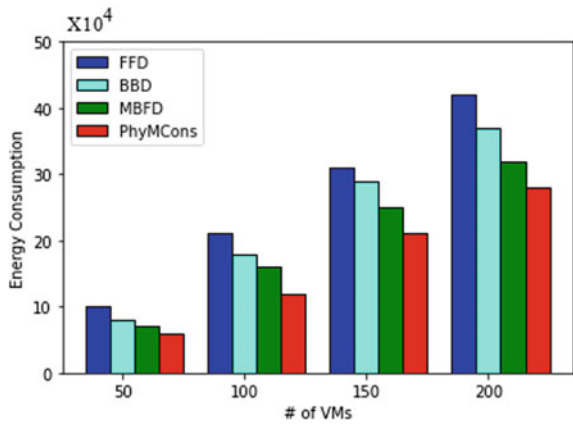
algorithm is better as compared to the FFD, BFD, and MBFD algorithms for the same number of VMs used.

8 Conclusion

One of the key advantages of virtualization technology for fog centers is physical machine consolidation. By consolidating virtualized computing environments to a smaller number of nodes, it allows for more efficient utilization of a pool of real equipment. Through the consolidation of physical machines, we can save money on energy, cooling, and administration. The use of live migration technology in fog centers brings up new possibilities for physical machine consolidation. This method allows for VM migration without interrupting their operation, allowing for dynamic VM replacement across a cluster of nodes. We can condense VMs into fewer servers when we notice VM's dispersion around the cluster without interrupting the

services operating on the VMs. This is accomplished in this study. In this paper, we implemented PhyMCons, a PM consolidation technique with two goals: reduce the servers used as well as reduce the migrations frequency. PhyMCons was compared to FFD, a widely used heuristic technique for bin-packing. The difference between our challenge and the bin-packing problem is that we consider the cluster's current state, i.e., the mapping among VMs and nodes, before performing the procedure. This data assists us in minimizing VM migrations along with minimizes the number of servers used.

References

1. Berl A, Gelenbe E, Di Girolamo M, Giuliani G, De Meer H, Dang MQ, Pentikousis K (2010) Energy-efficient cloud computing. Comput J 53(7):1045–1051
2. Galambos G, Woeginger G (1995) On-line bin packing—a restricted survey. Math Methods Oper Res
3. Goswami V, Patra SS, Mund GB (2012) Optimal management of cloud centers with different arrival modes for cloud computing environment. Int J Cloud Appl Comput (IJCAC) 2(3):86–97
4. Patra SS (2018) Energy-efficient task consolidation for cloud data center. Int J Cloud Appl Comput (IJCAC) 8(1):117–142
5. Rosenblum M, Garfinkel T (2005) Virtual machine monitors: current technology and future trends. Computer 38(5):39–47
6. Clark C, Fraser K, Hand S, Hansen JG, Jul E, Limpach C, Warfield A (2005) Live migration of virtual machines. In: Proceedings of the 2nd conference on symposium on networked systems design & implementation, vol 2, pp 273–286
7. Goswami V, Patra SS, Mund GB (2013) Dynamic provisioning and resource management for multi-tier Cloud based applications. Found Comput and Decis Sci 38(3):175–191
8. Horri A, Mozafari MS, Dastghaibyfard G (2014) Novel resource allocation algorithms to performance and energy efficiency in cloud computing. J SupCom 69(3):1445–1461
9. Goswami V, Patra SS, Mund GB (2012) Performance analysis of cloud with queue-dependent virtual machines. In: 2012 1st International conference on recent advances in information technology, pp 357–362
10. Kang J, Park S (2003) Algorithms for the variable sized bin packing problem. Eur J Oper Res 147(2):365–372
11. Song Y, Wang H, Li Y, Feng B, Sun Y (2009) Multi-tiered on-demand resource scheduling for VM-based data center. In: 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid. IEEE, pp 148–155
12. Li J, Zhao J, Li Y, Cui L, Li B, Liu L, Panneerselvam J (2015) iMIG: toward an adaptive live migration method for KVM virtual machines. Comput J 58(6):1227–1242
13. Naik M, Barik L, Kandpal M, Patra SS, Jena S, Barik RK (2021) EVMAS: an energy-aware virtual machine allocation scheme in Fog Centers. In: 2021 2nd International Conference for Emerging Technology (INCET). IEEE, pp 1–6

Particle Swarm Optimization-Based Photovoltaic Maximum Power Tracking Under Partial Shading Conditions: Performance Analysis

Raju Bhoyar and Sanjoykumar Mishra

Abstract Withdrawal of maximum power from PV array under variable atmospheric condition is difficult due to its nonlinear power characteristics. Power characteristics of PV array exhibit multiple peaks, and conventional maximum power point tracking algorithms are not able to locate the global peak power point. Due to this working deficiency of the conventional MPPT algorithms, output power of PV array could not be optimized under variable atmospheric conditions due to power mismatch and trapping of local minima under partial shading conditions. In this paper, comparative performance analysis of previous various research approaches of particle swarm optimization algorithm of maximum power point tracking is carried out on the basis of performance parameters and their contribution. This comparative performance analysis will contribute to help the researchers to locate the research gap and create a scope for improvement in the performance of PSO method.

Keywords Photovoltaic (PV) · Particle swarm optimization (PSO) · Maximum power point tracking (MPPT) · Partial shading conditions (PSCs) · Differential evolution (DE) · Global power point (GPP) · Maximum power point (MPP)

1 Introduction

The world has already seen the solar power as the cheap and pollution-free energy source to meet continuous need of electricity. However, cost of photovoltaic panels is still not affordable to the common users of electricity. Therefore, recently, denigrating the price of photovoltaic panels has turned in to crucial design study [1]. Uses of solar photovoltaic systems are being elevating progressively in the form of standalone systems and grid-connected systems. Basically, the output power of any PV module is decided by solar irradiance and temperature. Maximal power can be reduced from the PV source under consistent solar insolation. However, due to nonlinear output of PV source is a big challenge to locate global peak power point

R. Bhoyar (✉) · S. Mishra
Department of Electrical Engineering, G H Raisoni University, Amravati, India
e-mail: rajeevbhoyar@gmail.com

on output characteristics. Further, characteristics of PV array depict multiple peaks under non-uniform insolation and turned out to be complicated for tracking of global peak among the various local peak points [2].

PV system can be built up to meet the estimated load demand. However, PV system receives irregular solar irradiation due to partially shading of PV cells or even modules due to adverse environmental conditions, replica of trees, buildings, and other projected objects. Such situations are known as partial shading conditions of PV system which can results in serious impact on the power output of PV system. Bypass diodes are connected in PV system circuit to mitigate the effect of partial shading situation. PV module or array configuration decides the number of strength of the bypass diodes required in the PV modules [3].

For locating of optimal power point position on power characteristics of PV system, various MPPT techniques have been deduced by many researches, namely perturb and observe, hill climbing and incremental conductance. Most of developers of PV systems use these techniques. These approaches are simple to implement, but due to inadequate mind reasoning ability, these approaches could not able locate global peak under the partial shading situations [4].

Many researchers have been working on nature-inspired optimization algorithm for locating the global peak of PV characteristics. PSO is seen to be a simple coded algorithm to complex PV curves. Basically, in PSO algorithm, real problem lies in decisiveness of its working parameters and provides a feasible solution for early tracking of global maximum power point [5]. Practically, it is very uncertain to decide the particle velocity vector parameters for early tracking of global peak of PV curve in presence of the varied partial shading states. The guarantee of its practical feasibility in extracting the global maximum power peak depends of its proper selection of parameters.

Researchers are working on PSO algorithm involving most favorable parameter selection for reducing the search region [6]. Even though the PSO has untimely convergence which is its main pitfall, PSO can provide excellence in performance of PV system operations [7]. Most of the PV system uses perturb, and observe algorithm is under the normal insolation conditions to fetch maximum power; however, it slips to detect the universal maximal operating point under the partial shading conditions [8]. To improve the performance conventional PSO algorithm necessary modifications in the search velocity equations successfully incorporated to reduce unjustified movement of search particles to minimize the tracking time [9]. A nature-inspired optimization algorithms, namely "Differential Evolution and Particle Swarm Optimization" search procedures are used by PV industries in scheming maximal power point tracking algorithms of PV system [10].

In snow-covered PV module, single diode five parameter PV model is enhanced by the PSO algorithm [11]. Comparison of distinct particle swarm optimization constructed algorithms on the ground of convergence speeds, and shading designs indicate usefulness of its performance [12].

Improvement in conventional PSO search strategy for extracting of maximum power from grid-connected PV system is presented and found improved MPPT technique as compared to the conventional PSO technique [13]. Author elaborated detail

study of swarm intelligence algorithms to motivate researcher to carry out further research in computing techniques based on biologically based artificial intelligence [14]. Nature-inspired optimization algorithms can manage the issues related with management of high-dimensional data. Environmental conditions may affect the dataset of optimization process [15].

In this paper, comparative performance analysis of developmental stages in the algorithms of particle swarm optimization which are necessary for efficient tracking of maximum power from PV source in all the listed literature's in the reference section is carried out.

Apart from Sect. 1 as introduction which introduces the background and provides literature review, this paper is organized as follows—Sect. 2 describes the PV system model; Sect. 3 illustrates PSO technique; Sect. 4 put forward the previous research work on PSO algorithm approaches for PV MPPT, performance analysis of various PSO algorithms based on parameters contribution of PSO approaches and implementation of hardware and simulation, and discussion is carried out in Sects. 5 put together inferences of performance analysis as the conclusion part of the paper.

2   PV System Model

Basically, an accurate modeling of PV cell is necessary for designing the efficient PV system. Researchers worked on PV cell modeling to reduce the complexity in mathematical modeling and build up efficient PV cell model [16]. Practically, PV cell has a relatively low efficiency, and therefore, to obtain the improved performance, precise modeling of PV cell is needed. Mathematical modeling of PV cell under partial shading conditions by selecting suitable parameters is carried out to implement the PSO algorithm for tracing of global power point of a multi-peak PV characteristics system [17].

Figure 1 represents the PV cell model which is used for mathematical modeling of electrical characteristics equations of PV cell. The model incorporates a current
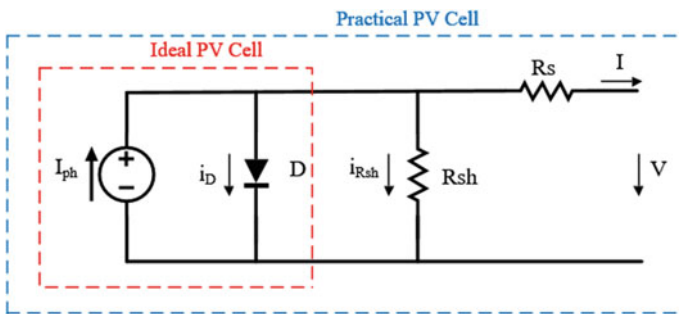
Fig. 1 Model of a single-diode PV cell [11]
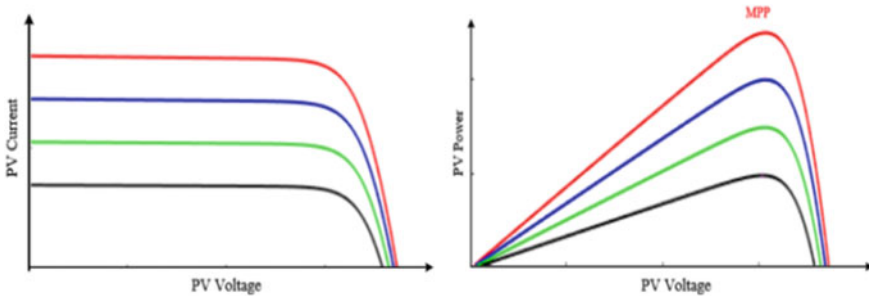
Fig. 2 *I–V* and *P–V* characteristics of PV panel at distinct environmental situations

source, diode, shunt, and series resistances as shown in Fig. 1. Current source denoted by I_{ph} represents the PV cell current which is created when photons from solar radiations falls on the cell. A shunted diode represents the PV cell. PV cell resistance is denoted by R_{sh} known as shunt resistance, which existed due to impurity of PV cell material. R_s represents resistances existed due to metallic junctions of PV cell and external circuit [10]. Double-diode model provides better accuracy under variation in ideality factor and saturation current in fitting of PV curves.

Figure 2 depicts current–voltage characteristics and power characteristics of PV panel at distinct environmental situations. These characteristics shows variation of PV voltage current and MPP at different irradiation and constant temperature.

The derived equation for PV characteristics of PV module is written as [11]:

$$I = I_{ph} - I_s\left[\exp\left(\frac{V + R_sI}{\alpha_o V_t}\right) - 1\right] - \frac{V + R_sI}{R_{sh}} \tag{1}$$

where I_s represent the reverse saturation current of diode [11], α_0 is the diode ideality factor, and $V_t = N_skT_c/q$ represents the PV module thermal voltage.

3 Particle Swarm Optimization

Solution to the nonlinear equations by optimization process was initialized by James Kennedy and Russell Eberhart using particle swarm methodology in 1948 and developed an intelligent optimization algorithm in 1995 [3]. The PSO algorithm works on social behavior of birds means the flocking movement of birds. In the particle swarm optimization method, each bird can be treated as a particle and movement of particle in a specific direction in search space as the bird takes the position during searching of food in the space.

The optimization begins with participation of each particle and further continues with the reporting of the local best particle, and global best particle is nominated as P_{best} and G_{best}. During search process, each particle adjusts its own place and velocity

Fig. 3 Movement of particle in the PSO process [2]
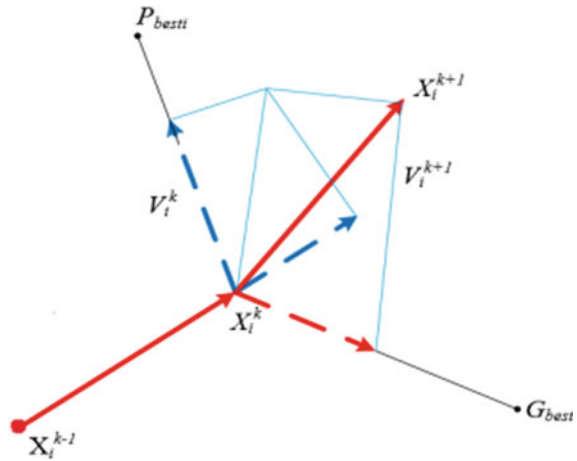
with reference to previous value of local best and global best positions. In search space, all particles can move in such a pattern that they are gradually shifting toward the best particle. Further, all particles moves continuously until a global solution is achieved [12]. During this optimization process, the particles move the search space in different directions as shown in Fig. 3 for first iteration.

Particle swarm optimization search procedure is uncomplicated and vigorous. It can be efficiently applied for optimizing a variety of functions, basically nonlinear functions.

It highly resides on hypothetical processes, like evolutionary programming. Being a simple, this algorithm requires specifications of the problem under study and a few parameters in view of obtaining the best solution to the study. Figure 3 depicts the velocity vector $v^k_{i,}$ which represents velocity of each particle moving in the search space.

PSO is described by the succeeding set of equations.

$$v_i^{k+1} = wv_i^k + c_1 r_1 p_{\text{best}} + c_2 r_2 g_{\text{best}} \tag{2}$$

$$s_i^{k+1} = s_i^k + v_i^{k+1} \tag{3}$$

$$P_{\text{best}i} = s_i^k \tag{4}$$

$$f(s_i^k) > f(P_{\text{best}i}) \tag{5}$$

where w is the weight coefficient, c_1 is cognitive coefficient, and c_2 social factor; r_1 and r_2 are the normalized random variables which uniformly lies between 0 to 1.

Equations (2) and (3) represent velocity and position vector of search particle. Next position of particle during search process is decided by itself previous best position

p_{best}, and the previous best position is achieved throughout the searching particles g_{best}. Equation (4) represents search particle best position, and the variable $p_{\text{best}i}$ reserves the best position of the ith particle. Equation (5) represents the optimizing function f that is maximized in each sequential cycle. The point g_{best} reserves the most desirable position attained among all the particles involved in the search process.

4   Previous Research Work on PSO

Literature search from previous research work on maximum power tracking using particle swarm optimization techniques is carried out. Many researchers have found worked on significant approaches of PSO which have been presented as listed in references. Key features of most of previous work are put forward as below.

4.1   *Adaptive Perceptive PSO Algorithm*

In this article, author proposed the algorithm which overcomes the disadvantages of perceptive PSO algorithm for optimum power point tracking of PV modules subjecting to partial shadowing. In adaptive perceptive PSO algorithm particle local best position in the current iteration improves its performance, and its personal best position is streamlined in next iteration to locate precisely the maximum power point of a PV module [1].

4.2   *A Novel MPPT PSO Algorithm*

In this article, researcher developed a simple multidimensional search-based PSO algorithm to control several PV arrays using suitable voltage and current sensors. To prove its novelty, experimental comparison with various tracking techniques was carried out [2].

4.3   *A Modified PSO Algorithm*

In this article, author proposed use of direct duty cycle control along with conventional PSO algorithm to raise tracking veracity of maximum power point of PV system. In modified PSO algorithm, steady-state oscillations can be dropped down to zero by maintaining the constant value of duty cycle [18].

4.4 A Modified PSO Algorithm

Author extended the work presented which is discussed in sub-Sect. 4.3. As per authors' perception, system design guidelines and practical design considerations were not mentioned by authors [18] in their research paper. In this paper, author has implemented design procedure through hardware limitations and modified the PSO algorithm [3].

4.5 A Deterministic PSO Algorithm

Author introduced an idea to exclude the use of the random variable in the velocity equation of conventional particle swarm optimization to reduce the search region [4]. Due to this optimization process become simple and trapping of search agent in the vicinity of global peak is possible in presence of partial shading situations [4].

4.6 Improved PSO Algorithm

In this algorithm, author focused on more practical facts, namely any PV cell failure and shading circumstances, which can reduce the output power of PV module in view of tracking time, response, and efficiency [19].

4.7 A Modified PSO Algorithm

In this article, author developed modified PSO algorithm which focused the initial value selection for maximum power point tracking. For fast convergence, this paper assimilated useful approach for determination of initial value of duty cycle with reduced power fluctuations at global peak of PV Curve [20].

4.8 A Simplified Accelerated PSO Algorithm

Author proposed selection of search coefficients values to simplify the search process and to reduce the computational needs of PSO algorithm [21]. Testing of this modified PSO, algorithm under different shading conditions is carried out to locate the global peak.

4.9   A Forward–Backward PSO Algorithm

In this article, author presented PSO algorithm, which start the initial search from both forward and backward direction, and best initial estimates obtained are used in the further search process to narrow down the tracking time [22].

4.10   Electrical Wiring Reconfiguration of PV Array-Based PSO Algorithm

In this article, author presented effect of changing the electrical connections of PV array on output power of PV array under PSCs using PSO method. In this technique, the electrical connections are changed keeping the physical location of the modules remains unchanged [23].

4.11   An Improved PSO Algorithm

Author proposed modified version of a conventional PSO algorithm to solve the practical problems and to locate the changes in shading patterns of PV array working under PSC. Proposed version incorporates the manipulations' in the values of the velocity equation search agents' coefficients and weight factors [24].

4.12   Optimal Parameters Selection of PSO Algorithm

In this PSO algorithm, author considered the solar array physical arrangement and designed scheme and parameters of buck converter for best selection of PSO algorithm parameters' for withdrawal of maximum power of the PV system. In this work, author used a novel new method for selection of buck converter parameters [6].

4.13   PSO Algorithm Based on Irradiation and Temperature Measurement

In this research work, author presented the PSO algorithm based on temperature measurements to increase the output power of PV array subjecting to partially shaded conditions. This approach has reduced the complexity in framing of fitness function and their by increasing convergence speed of the algorithm [25].

4.14 Enhanced Leader PSO Algorithm

Author presented enhancing of the leader particle using successive mutation method helping the proper convergence of leader particle to the global MPP point. Author has used five-staged successive mutation strategy for enhancing the leader particle and obtained more accurate parameters of PV model [7].

4.15 Modified Particle Velocity-Based Algorithm

In this approach, author proposed a modification in the weight factor and coefficients of velocity equation of PSO by introducing the adaptive values to them. This algorithm reduces the search time in search space and results in reduction in the oscillations around the global point [26].

4.16 High-Speed PSO Algorithm

In this algorithm, researcher presented sampling of PV curve using high resolution analog-to-digital converter to obtain PV system terminal voltage corresponding to peak power point. This fast-tracking approach works effectively both in uniform and irregular insolation conditions with enhancing the performance parameters for MPPT of PV system [8].

4.17 Hybrid PSO Algorithm

Researcher proposed hybrid algorithm using the conventional adaptive perturb and observe and particle swarm optimization techniques [9]. Within algorithm, search space is minimized by implementing the search-skip-judge mechanism. Author tested the proposed algorithm against the other methods, namely cuckoo search, modified incremental conductance, the old approach of search-skip-judge, and the hybrid PSO.

4.18 Novel Initialization Scheme-Based PSO Algorithm

In this scheme, author proposed initialization of populations using a novel center-based Latin Hypercube Scheme [10]. Author carried out relative analysis of results of both differential evolutionary and particle optimization algorithms and figured out

best parameter settings of both algorithms for optimum power position tracking of PV system.

4.19 A Model-Based PSO Algorithm

In this PSO approach, author proposed simultaneous determination of unknown values of PV model parameters with the use of the magnitude of temperature and insolation present in the snow [11]. In variable snow situation, author used PSO evolutionary algorithm for calculation of instantaneous values of proposed model parameters of PV modules T.

4.19.1 Levy Flight-Based PSO Algorithm

In this article, author proposed elimination of steady-state power oscillation by updating the step size of velocity by Levy flight of PSO for GPP tracking of PV system under PSCs. In this algorithm by updating the step size of velocity of particle by Levy flight, required iterations reduce to fewer one in order to increase convergence speed with low steady-state oscillations [27].

4.19.2 Variant-Based PSO Algorithm

In this paper**,** author presented relative analysis of conventional PSO approach and three different approaches of PSO. These three PSO approaches are works on timely variation of velocity vector coefficients, namely inertial weight coefficient and acceleration coefficient. The performance is compared in accordance with convergence rate under dissimilar partial shading patterns [12].

5 Performance Analysis of Previous PSO Works

Critical comparative performance analysis of previous research work of on various approaches of PSO of PV maximum power tracking under different partially shaded situations is carried out. The relative performance analysis is tabulated for highlighting the performance parameters named as sensed parameters, algorithm complexity, tracking time, tracking efficiency, dynamic response steady-state oscillations, and PSC track ability, and contribution of various previous PSO approaches is discussed in this paper. This analysis revealed uniqueness and significance of each PSO maximum power tracking technique.

5.1 Analysis Discussion

Various PSO algorithms published from the year 2010 to 2020 focused on control parameter using suitable hardware and simulation platform to improve the PV performance. Table 1 indicates performance analysis of PSO approaches on the basis of various working parameters and Table 2 based on contribution made by them and implementation of hardware and software simulation. Some researchers had focused on rearrangement of PV module configuration while others on proper selection of PSO parameters to extract the maximum power under partial shading conditions. Performance analysis reveals that with the achievement of less tracking time and reduced steady-state oscillations, tracking efficiency decreased somehow.

PV electrical wiring-based PSO technique requires only single sensor but suffers to improve other performance parameters. Most of the PSO algorithms used V&I sensing parameters and found better improvement if PV performance as compared to the irradiation and temperature input parameters PSO algorithms. Each algorithm has specific contribution under PSCs of PV system to reduce the temperature effect on PV cell, reduction in cost of control system, improved convergence rate, tracking efficiency, reduction in computational complexity, tracking time, system cost, applicability to large PV system.

6 Conclusions

In this study, relative performance analysis of PSO-based maximum power tracking algorithms which are presented in the Sect. 4 is worked out. Comparative performance analysis is summarized in Tables 1 and 2, and an attempt is made to highlight performance parameters and contributions of the different PSO approaches fetch the maximum power from PV system. It is observed that PSO is simple to implement and has an ability to fetch optimal power from PV array with various partly shaded situations with reduced computational complexity. It also reveals that each approach has its own uniqueness and significance. This comparative performance analysis will contribute to help the researchers to locate the research gap and create a scope for improvement in the performance of PSO method to increase in power dispatch by any type of PV system subjecting to all type environmental conditions.

Table 1 Performance analysis of previous works PSO approaches on PSO parameters

PSO MPPT algorithm	Sensed parameters	Algorithm complexity	Tracking time	Tracking efficiency	Dynamic response	Steady-state oscillations	PSC track ability
Adaptive perceptive PSO	V & I	Easy	1–2 s	97.7%	Good	Less	High
A novel PSO	V & I	Easy	2 s	NM	Good	Less	High
A modified PSO	V & I	Easy	Less	NM	Good	Zero	High
A modified PSO	V & I	Easy	Less	99.5	Good	Less	higH
A deterministic PSO	V & I	Easy	Less	>99.5%	Good	Less	High
An Improved PSO	V & I	Easy	15 ms	NM	Good	NM	High
A modified PSO	V & I	Easy	<1 s	NM	Good	Almost zero	High
A simplified accelerated PSO	V & I	Easy	Less	NM	Good	Less	High
A forward–backward PSO	V & I	Easy	3.6 s	99.99	Good	Less	High
Electrical wiring of PV array-based PSO	I	Average	NM	NM	Good	NM	High
An improved PSO	V & I	Easy	1.9 s	99.91	Good	Less	High
Optimal parameters' PSO	V&I	Easy	0.28 s	99.85	Good	Less	High
Irradiation and temperature PSO	V, I, Temp. and Irradn.	Average	Less	NM	Good	NM	Average
Enhanced leader PSO	V & I	Easy	NM	NM	Good	NM	High
Modified particle velocity PSO	V & I	easy		99.95	good	NM	High

(continued)

Table 1 (continued)

PSO MPPT algorithm	Sensed parameters	Algorithm complexity	Tracking time	Tracking efficiency	Dynamic response	Steady-state oscillations	PSC track ability
High-speed PSO MPPT	*V & I*	easy	0.03 s	NM	good	almost zero	High
Hybrid PSO	*V & I*	easy	2.2 s	NM	good	less	High
Novel initialization-based PSO	*V & I*	easy	less	NM	good	less	High
Model-based PSO	*V & I*	easy	NM	NM	good	NM	NM
Levy flight-based PSO	*V & I*	easy	0.3 s	99.85	good	less	High
Variant based PSO	*V & I*	easy	less	NM	good	less	High

NM Not mentioned in the research paper, *V&I* voltage and current, *Temp.* temperature, *Irradn.* irradiation

Table 2 Performance analysis of previous works PSO approaches on contribution and implementation of hardware and simulation

PSO MPPT algorithm	Ref. year	Control parameter	DC–DC converter	Controller	Contribution
Adaptive perceptive PSO	2010	Duty cycle	Buck	MSP430FG4618	Applicable to large-scale PV system
A novel PSO	2011	Duty cycle	Boost	DSP TMS320C32	Reduction in overall the cost of the control system
A modified PSO	2012	Duty cycle	Buck–boost	TMS320F240 DSP	High convergence speed with high tracking efficiency
A modified PSO	2012	Duty cycle	Boost	DSC PIC33FJ16GS502	For centralized-type PV system operating under PSC
A deterministic PSO	2013	Duty cycle	Buck–boost	TMS320F240 DSP	Improvement in tracking speed and efficiency
Improved PSO	2015	Duty cycle	Boost	PIC181F8720	Reduction in tracking time and improved MPPT algorithm
A modified PSO	2015	Duty cycle	Boost	Arduino	Parameter initialization and identifying the optimal duty cycle
A Simplified accelerated PSO	2016	Duty cycle	Boost	PIC16F877A	Computational need is reduced hence results in speedy search process
A forward–backward PSO	2017	Voltage	NM	Matlab simulation	Improved search process to reduce tracking time
Electrical connections of PV array-based PSO	2018	Current	NM	Matlab simulation	Critical cost analysis, energy saved, and income generation

(continued)

Table 2 (continued)

PSO MPPT algorithm	Ref. year	Control parameter	DC–DC converter	Controller	Contribution
An improved PSO	2017	Duty cycle	SI-SEPIC	OPAL-RT OP5060	For centralized-type PV system operating under PSC
Optimal parameters' selection PSO	2017	Duty cycle	Buck	Matlab simulation	Buck converter design and parameter selection
Irradiation and Temperature PSO	2016	Duty cycle	Boost	Dspic4011	Temperature effect on PV cell
Enhanced leader PSO	2018	Current	NM	Matlab simulation	Premature convergence problem is alleviated
Modified particle velocity PSO	2018	Duty cycle	Boost	TMS320F28069	For tuning of velocity equation parameters
High-speed PSO MPPT	2018	Duty cycle	Boost	PI	All type of PV array at any type of weather situation
Hybrid PSO	2018	Duty cycle	Buck–boost	TMS320F240 DSP	Reduced unnecessary movement of particles
Novel initialization-based PSO	2018	Duty cycle	Boost	Matlab simulation	Convergence rate and solution accuracy. and effective population initialization
Model-based PSO	2019	Duty cycle	Boost	Matlab simulation	To improve PV performance under snow conditions
Levy flight-based PSO	2020	Duty cycle	Boost	Matlab simulation	No tuning of velocity equation parameters requires
Variant-based PSO	2020	C_1, C_2, w	NM	Matlab simulation	Reduction in computational complexity and tracking time

References

1. Chowdhury S, Saha H (2015) Maximum power point tracking of partially shaded solar photovoltaic. Arrays Solar Energy Mater Solar Cells 94:1441–1447
2. Miyatake M, Veerachary M, Toriumi F, Fujii N, Ko H (2011) Maximum power point tracking of multiple photovoltaic arrays: a PSO approach. IEEE Trans Aerosp Ind Electron Syst 47:367–380
3. Liu Y, Huang SC, Huang J, Liang W (2012) A particle swarm optimization-based maximum power point tracking algorithm for PV systems operating under partially shaded conditions. IEEE Trans Energy Convers 27:1027–1035
4. Ishaque K, Salam Z (2013) A deterministic particle swarm optimization maximum power point tracker for photovoltaic system under partial shading condition. IEEE Trans Ind Electron 60:3195–326
5. Seyedmahmoudian M, Mekhilef S, Rahmani R, Yusof R, Shojaei A (2014) Maximum power point tracking of partial shaded photovoltaic array using an evolutionary algorithm: a particle swarm optimization technique. Renew Sustain Energy 6:1–13
6. Obukhov S, Ibrahim A, Diab A, Al-Sumaiti A, Aboelsaud R (2017) Optimal performance of dynamic particle swarm optimization based maximum power trackers for stand-alone PV system under partial shading conditions. IEEE Open Access 1–18
7. Rezaee Jordehi A (2018) Enhanced leader particle swarm optimisation (ELPSO): an efficient algorithm for parameter estimation of photovoltaic (PV) cells and modules. Sol Energy 159:78–87
8. Sankar S, Mohanty M, Chandrasekaran K, Simon S, Sood Y (2018) High-speed maximum power point tracking module for PV systems. IEEE Indus Power Electron 66:1119–1129
9. Kermadi M, Salam Z, Ahmed J, Berkouk M (2018) An effective hybrid maximum power point tracker of photovoltaic arrays for complex partial shading conditions. IEEE Trans Indus Electron 66:6690–7000
10. Khan F, Sunbul A, Ali MY, Abd El-Gawad H, Rahnamayan S, Sood V (2018) Maximum power point tracking in photovoltaic farms using DE and PSO algorithms: a comparative study. IEEE Congress Evolut Comput (CEC) 27
11. Khenar M, Hosseini S, Taheri S, Cretu A, Pouresmaeil E, Taheri H (2019) Particle swarm optimisation-based model and analysis of photovoltaic module characteristics in snowy conditions. IET Renew Power Gener 13:1950–1957
12. Iqbal D, Ahmad T, Pervez I, Malick HK, Sarwar A, Tariq M (2020) Performance of PSO based variants in tracking optimal power in a solar PV based generation system under partial shading condition. Smart Sci 8:1–13
13. Fernando M, Sérgio A, Fábio R, Leonardo P, Vinícius D, Leonardo B (2016) Grid-tied photovoltaic system based on PSO MPPT technique with active power line conditioning. IET Power Electron 9:1180–1191
14. Acharjya D, Kauser A (2015) Swarm intelligence in solving bio-inspired computing problems: reviews, perspectives, and challenges. Nat Insp Comput 1–3:87–110
15. Khare N, Devan P, Chowdhary C, Bhattacharya S, Singh G, Singh S, Yoon B (2020) Spider monkey optimization and deep neural network hybrid classifier model for intrusion detection. Electronics 9(4):692
16. Dileep G, Singh S (2017) Application of soft computing techniques for maximum power point tracking of spv system. Sol Energy 141:182–202
17. Liu J, Li J, Wu J, Zhou W (2017) Global MPPT algorithm with coordinated control of PSO and INC for rooftop PV array. IET Renew Power Gener 13:778–782
18. Ishaque K, Salam Z, Amjad M, Mekhilef S (2012) An ımproved particle swarm optimization (PSO)–based MPPT for PV with reduced steady-state oscillation. IEEE Trans Power Electron 27:3627–3637
19. Chao K, Lin Y, Lai U (2015) Improved particle swarm optimization for maximum power point tracking in photovoltaic module arrays. Appl Energy 258:609–618

20. Venugopalan R, Krishnakumar N, Sudhakarbabu T, Sangeetha K, Rajasekar N (2015) Modified particle swarm optimization technique based maximum power point tracking for uniform and under partial shading conditions. Appl Soft Comput 34:613–624
21. Rajendran S, Srinivasan H (2016) Simplified accelerated particle swarm optimisation algorithm for efficient maximum power point tracking in partially shaded photovoltaic systems. IET Renew Power Gener 10:1340–1347
22. Ishaque K, Salam Z, Amjad M, Mekhilef S (2017) A forward backward PSO based MPPT algorithm for PV system under partial shading condition. Int J Control Theory Appl 10:33–43
23. Babu T, Ram J, Dragicevi T, Miyatake M, Blaabjerg F, Rajasekar N (2018) Particle swarm optimization based solar PV array reconfiguration of the maximum power extraction under partial shading conditions. IEEE Trans Sustain Energy 9:74–85
24. Dileep G, Singh S (2017) An improved particle swarm optimization based maximum power point tracking algorithm for PV system operating under partial shading conditions. Sol Energy 158:1006–1015
25. Sridhar R, Jeevananthan S, Pradeep V (2016) Particle swarm optimization maximum power tracking approach based on irradiation and temperature measurements for partially shaded photovoltaic system. Inte J Amb Energy 685–693
26. Sen T, Pragallapati N, Agarwal V, Kumar R (2018) Global maximum power point tracking of PV arrays under partial shading conditions using a modified particle velocity-based PSO technique. IET Renew Power Gener 12:555–564
27. Motamarri R, Nagu B (2020) GMPPT by using PSO based on Lévy Flight for photovoltaic system under partial shading condition. IET Renew Power Gener 14:1143–1155

Electrical Resistivity Investigation for Dam Site

D. S. Aswar, P. B. Ullagaddi, and S. D. Ambadkar

Abstract The resistivity surveys give the variation of electrical resistivity with the depth. The vertical electrical resistivity sounding (VES) was carried out at Bhama Askhed dam site approximately aligned with the dam alignment on downstream to generate additional data to complement existing geological data. The VES interpretations carried out using resistivity inversion (IR) and inverse slope method (ISM) to generate resistivity interpretations for part of modelled area. The field resistivity data is interpreted to obtain the true resistivity and the thickness of the ground layers. The electrical resistivity (ER) cross sections showing the variation of resistivity with the depth are also prepared. The findings of resistivity survey have complemented the available exploration data. The study has also helped to identify the occurrence of low-resistivity zones, which was missed earlier due to inadequate depth of geological investigation. Low-density of dam foundation substratum towards spillway area where the high seepage has occurred leading to EDA failure has been investigated.

1 Introduction

Geophysical methods may be used as complementary tool for site investigation. The resistivity method is used to study the horizontal and vertical variations in the subsurface electrical properties. It can also contribute as a tool to optimize extent of

D. S. Aswar (✉)
Sinhgad College of Engineering, Pune 411041, India
e-mail: dsaswar.scoe@sinhgad.edu

P. B. Ullagaddi
S.G.G.S. Institute of Engineering and Technology, Nanded 432606, India

S. D. Ambadkar
G H Raisoni University, Amravati 444701, India
e-mail: swati.ambadkar@ghru.edu.in

investigation required by identifying the locations of anomalies for further investigations. The electrical anomaly of substratum is mainly because of variations in the rock resistivity. The verified results of geophysical survey can be extended to areas of spare geological data. Geophysical methods have been extensively used in dam investigations, as well as in the assessment of existing dam structures.

2 Study Area Bhama Askhed Irrigation Project

The Bhama Askhed Irrigation Project (18° 15′N, 73° 43′E), is situated on the River Bhama, a right-bank tributary of River Bhima (Krishna Basin), at village Waki, Taluka- Khed, Pune, Maharashtra (India). The project consists of 51.125 m high and 1425 m long earthen dam with gross storage capacity at FRL of 230.47 MCM and 55.5 m long ogee side spillway. During 2005, heavy seepage accompanied with the dynamic spillway discharge forces, uplifted and displaces, thirteen concrete panels (7 × 11 × 0.30 m each) of flip bucket type, energy dissipating assembly (EDA). The structural failure of the EDA portion has occurred due to excessive seepage and associated uplift pressure [1, 2]. The discussions here are mostly confined here to analyse and interpret geological conditions leading to EDA failure.

3 Vertical Electrical Sounding

Proportion of current flowing beneath subsurface depth is a function of current electrode separation AB. High electrode separations is, however, limited by difficulty of working with lengthy cable and space availability on site [3]. The sounding locations were chosen considering the availability nearly horizontal, workable space for electrode spreading. The site topography has limited the depth of geophysical exploration due to limited scope of electrode separation. The tail channel area is with rocky outcrop and deep-water gullies making it difficult for establishing electrode contact points. The main purpose of vertical electrical sounding is to deduce the electrical resistivity variation with depth and thus to interpret the geological subsurface characteristics. Considering the advantages and the suitability, the Wenner array was selected for VES [4]. The electrical resistivity survey was carried out with the standard Wenner electrode array with current electrodes (AB) spacing varying from 1.5 to 200 m. The maximum current electrode separation used was AB = 200 m, with AB/3 = 65 m. Electrical resistivity survey was carried out along four geoelectric traverses based on available space, as geoelectric profile-I, profile-II, profile-III and profile-IV to obtain a possible coverage of the study area. Table 1 presents details of geoelectric traverses conducted during the field survey.

VES was conducted at 25 VES stations (VES_1 to VES_25) distributed along these traverses (Fig. 1). The profiles were orientation approximately parallel to dam alignment on downstream, with combined length of 1450 m.

Table 1 Details of geoelectric traverses

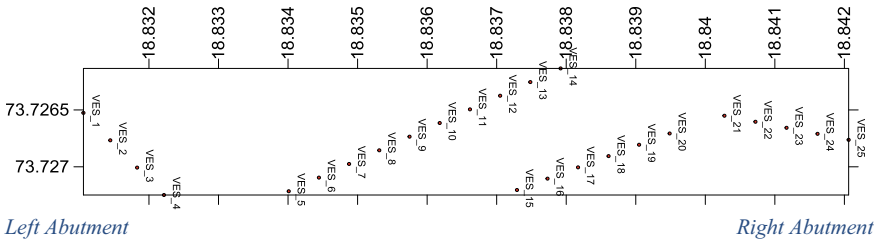| Traverse lines	VES stations	Fore bearing	Length (m)
Profile-I	VES_1 to VES_4	33° 30′	250
Profile-II	VES_5 to VES_14	12°	550
Profile-III	VES_15 to VES_20	9° 30′	350
Profile-IV	VES_21 to VES_25	9° 30′	300

Fig. 1 VES stations and geoelectric traverse location

The spacing between the VES stations was 50 m. Number of profiles and number of VES stations in each profile were selected based on available working space, accessibility and field conditions.

3.1 Interpretation Methods

VES data yields apparent resistivity for various electrode separations. This data is used to find true resistivity and depths of subsurface formations. The interpretation of resistivity sounding data can be done in two stages,

1. Processing the data to get the geoelectric parameters in terms of resistivities and depths/thicknesses of formations.
2. Geoelectric parameters interpretation for the nature of subsurface formations based on geological understanding and correlative studies [5].

Qualitative Interpretation. A qualitative method indirectly displays the information of the geological structure and the geological variations over the study area. The qualitative resistivity interpretation carried out for study consists of observation of the field sounding curve, isoapparent electrical resistivity maps, apparent resistivity profiles, pseudo sections and resistivity cross sections.

**Quantitative Interpretation**. Field resistivity data was interpreted using inverse slope method (direct method) [5] and automatic iterative inversion approach [6] to obtain the geoelectrical section parameters, i.e. layer thickness and resistivity values. The geoelectrical section parameters are obtained by plotting the sounding curves on natural scale in inverse slope method (ISM). Whereas, the sounding curves are plotted

on double logarithmic scale in resistivity inversion technique (IR). The true resistivity obtained was used to generate iso-resistivity contour, thickness map and resistivity profiles using SURFER software. The results are interpreted by correlating layer resistivity and corresponding formation thickness with the nearest borehole data.

Resistivity Interpretation by Resistivity Inversion (IR) Method. The vertical electrical resistivity soundings were interpreted using inversion techniques to obtain true subsurface resistivity, using computer-based inversion using software such as IPI2Win and EarthImager [7]. The inversion method minimizes the difference between the modelled and measured apparent resistivity. IPI2Win utilizes the geological concepts and data along with the resistivity observations, instead of only direct sounding curves inversion. Thus, providing equivalent resistivity solutions better suited to both geophysical data (i.e. the least-fitting error) and geology. Output consist of, resistivity curve, resistivity-depth table, pseudo cross-section and resistivity cross-section. The IPI2Win software suggests a best-fitting model for the initial interpretation for input sounding point data. The degree of uncertainty of the computed model parameters and its fit with standard curve are expressed in terms of fitting root mean square (RMS) error. A number of inversions reproduce the layer resistivity and the corresponding layer thickness until the model parameters of the VES curve are resolved with the fitting error. Model trial involves altering the number of layers by means of splitting or joining them and changing the properties of the layers. Layer boundary elevations are shown and calculated relative to surface elevations. Fix flags allow the user to fix parameters for the inversion calculations. The nature and distribution of different lithologic layers evidence by depth wise variation of resistivity values. The simulated results of the 25 VES stations have disclosed the existence of 4–6 geoelectric layers.

Resistivity Interpretations by Inverse Slope Method (ISM). ISM was used for quantitative interpretation of the vertical electrical sounding using Wenner configuration. It gives reasonably dependable depth calculations under most of the conditions [5, 8]. The inverse of apparent resistance obtained ($1/R$) is plotted against the Wenner electrode separation '$a = AB/3$' on a natural graph to detect layers at locations under consideration. Near surface, very thin layers are suppressed in inverse slope method; on the other hand, the effect of thinner deep layer is brought out clearly with ISM. With the adequate data density, ISM is able to use linear plotting to trace even thin layers buried at great depths. The ISM has enabled to distinguish distinct sublayers. A single lithological layer identified using the inversion technique could be separated into two or more sub layers by the ISM. This provided an additional advantage to use ISM in this case study.

Fig. 2 Resistivity sounding
curves (Profile-IV)
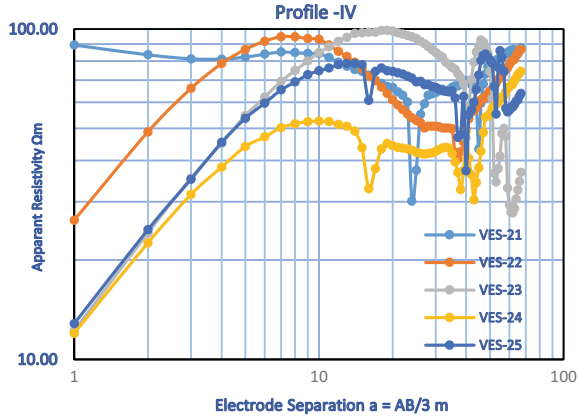
3.2 Geophysical Investigation Analysis—Sounding Curves Interpretations

The apparent resistivity, 'ρa' values were plotted against the electrode spacing ($a = AB$/3) on a double-logarithmic scale to obtain the resistivity sounding curve [9]. Fig. 2 represents such resistivity sounding curves for geoelectric traverse line along Profile-IV.

Observations from these sounding curves reveals that the profile-IV showing resistivity anomalies with depth. These low-resistivity zones in profile-IV towards spillway section around 40 m may be corresponding to low-density formations. The continuous rise in the sounding curve towards high-resistivity values indicates presence of a massive hard rock. However, the resistivity values decrease around 40 m for Profile-I and for Profile-IV and at 30 m for Profile-II and for Profile-III. These descends in resistivity values correspond to low-resistivity weaker zones at respective approximate depths. However, all the profiles marked increase in apparent resistivity values towards the end.

3.3 Isoapparent Resistivity Map

The qualitative interpretation involved isoapparent electrical resistivity maps and geoelectrical pseudo sections. The apparent resistivity value ($\rho 1$, $\rho 2$, $\rho 3$) of the individual layer was used for the preparation of isoapparent resistivity maps using Surfer software. These maps reflect the lateral variation of apparent resistivity over a horizontal plane at a depth of interest. Assuming the ideal depth of investigation for the Wenner configuration is approximately equal to the electrode spacing "a" or 1/3 of current electrode spacing (AB), the isoapparent electric resistivity maps were constructed at $AB = 102, 117, 135, 150, 168, 183$ and 201 m. These maps reflected the

lateral variations of the electric resistivity at an approximate depth of about 30, 35, 40, 45, 50, 55 and 60 m as shown in Fig. 3. The right and left abutment (bank) directions are assumed as looking towards the dam upstream, i.e. from downstream to upstream. Typically, all the isoapparent electric resistivity map at approximate depth up to 60 m showing the consistent low-resistivity values even with the increasing depth towards the northern area (around VES_23 and VES_24) towards the right bank spillway section. However, the resistivity slightly improves towards extreme Northern area after the depth of 55 m. These indicates the presence of uniform bedrock towards the left bank, i.e. Southern part. The apparent resistivity profile, Fig. 4, shows the
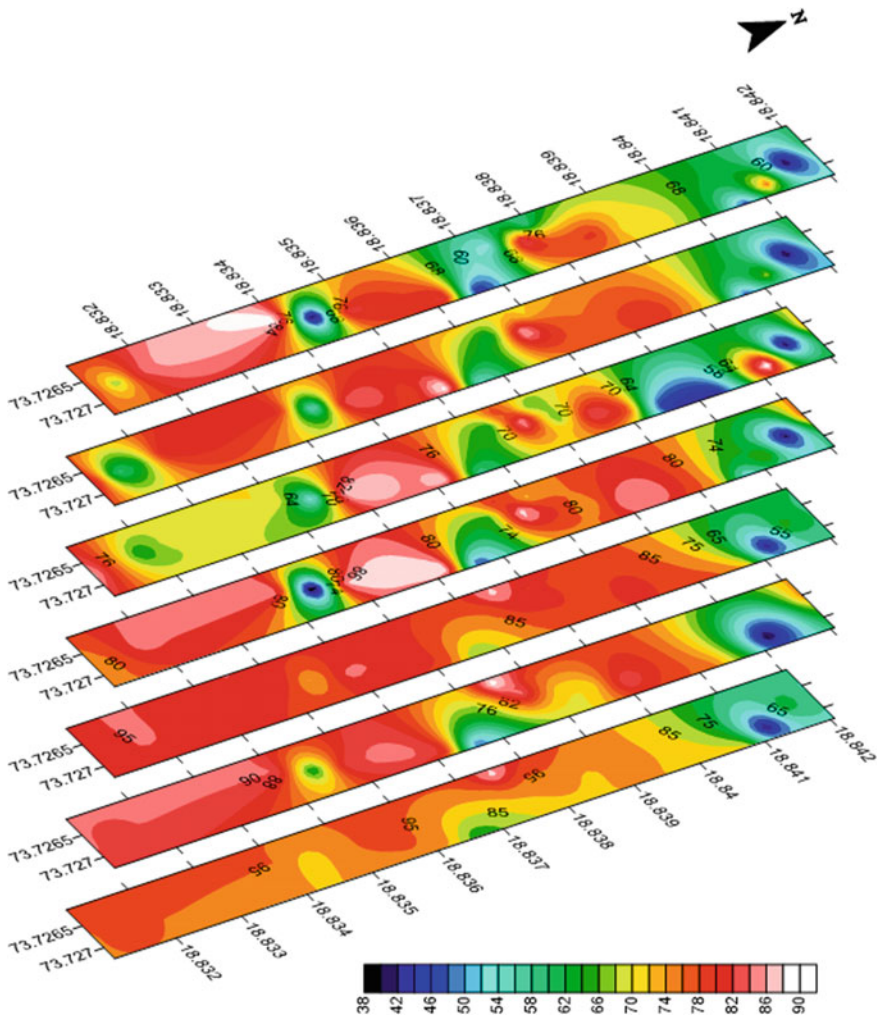

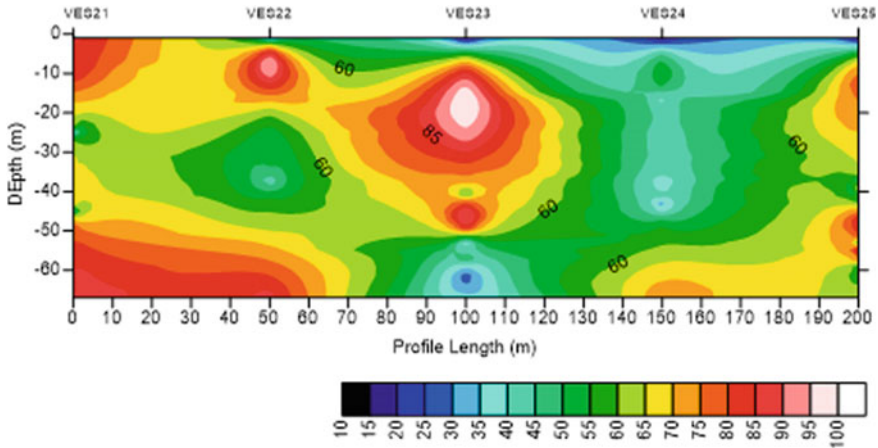**Fig. 3** Stacked isoapparent resistivity map

Fig. 4 Apparent resistivity Profile-VI (VES_21_25)

variation of apparent resistivity over the depth of substratum; the results are similar to pseudo resistivity cross sections obtained in inversion resistivity method. These apparent resistivity profiles are qualitatively easy to interpret compared to pseudo resistivity sections. Based on apparent resistivity, variation over the depth different lithological zones has been identified. The apparent resistivity profiles along the VES traverse were plotted to reflect qualitative geologic setting, along these profiles using Kriging data interpolation.

The obtained apparent resistivity values are low (40–50 Ω m) at the depth around 35–40 m (VES_22), 55–60 m depth (VES_23) and at depth 35–40 m (VES_24 and VES_25), at the Northern soundings towards spillway areas. These sounding stations show consistent low-density, saturated zones at the greater depths preceded by initial high density, high-resistivity area. Profile-IV is indicating the likely low-density trends towards spillway area where the high seepage has occurred leading to EDA failure.

3.4 Interpretations of the Inversion Method

2D electrical resistivity pseudo cross section (apparent resistivity) and resistivity cross section (true resistivity) were constructed based on VES data (VES_1 to VES_25) along Profile-I, Profile-II, Profile-III and Profile-IV (Fig. 5.)

These provided a simple image, for qualitative interpretation with spatial variation of apparent (pseudo cross section) and true subsurface resistivity information [10].

The detected layers at VES stations show a wide range of resistivity values with the last (deepest) layer showing very high resistivity normally for all the sounding points. However, VES stations VES_8 (29.8 Ω m), VES_9 (37.9 Ω m), VES_10
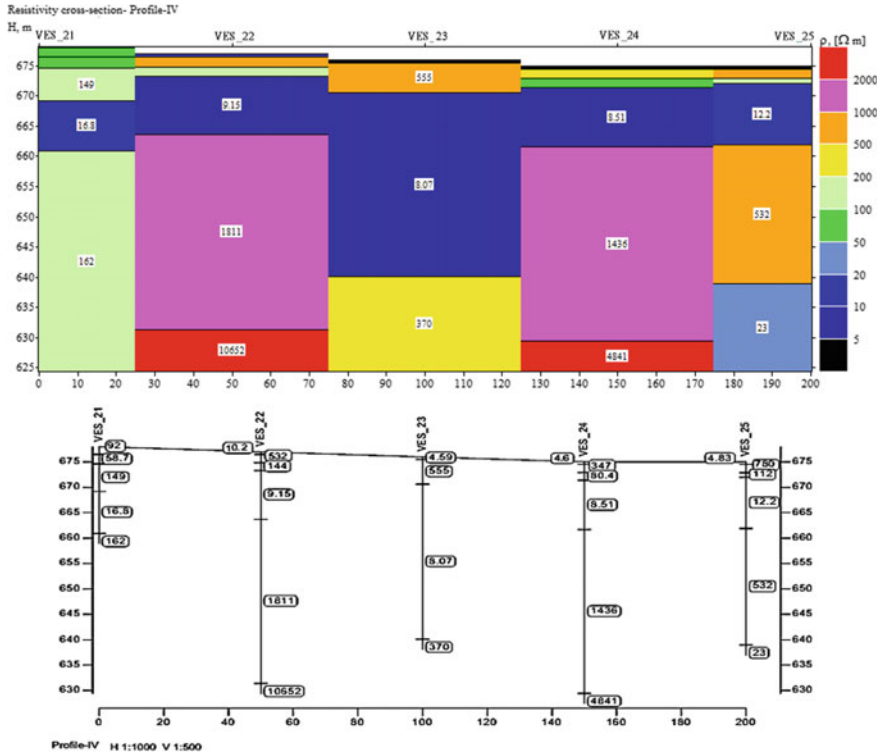
Fig. 5 2D electrical resistivity and geoelectric cross sections along the profiles III and IV

(17.4 Ω m), VES_19 (7.06 Ω m), VES_20 (44.7 Ω m) and VES_25 (23 Ω m) are terminating into low-resistivity layers and needs further investigation for verification of bedrock condition.

3.5 Inverse Slope Method Interpretations

Inverse slope method has enabled in distinguishing the sub layers, which correlate well with bore logs. The resistivity data was analysed, and ISM interpretations were carried out for all VES station (VES_1 to VES_25). Both IR and ISM interpretations, based on their methodological approaches, have given slightly variable results.

4   Result Comparison for IR and ISM

A. Layer thickness comparison for IR and ISM—In general, there was no specific trend regarding obtained layer thickness for VES stations using IR and ISM, however, for most of the cases, ISM is giving greater thickness for top layers compared to IR method. In IR method, the lower layers get compressed on logarithmic scale. In general, IR method shows a greater number of high resistivity low-thickness layers, which is the limitation of this method. On the natural scale ISM, decompress the layers at greater depth, identifying the better layer resistivity variation at greater depth as compared to IR method. Thus, giving a greater number of layers at greater depth.

B. Absolute Resistivity Comparison for IR and ISM—In general, ISM and IR layer resistivity values show good agreement for top layer. VES_1, VES_18 compares well for both ISM and IR, whereas VES_2, VES_4, VES_6, VES_7, VES_13, VES_16, VES_17, VES_19, VES_20, VES_21 show appreciable correlation for top layers. IR method in general is giving very high-resistivity value for the last layer with infinite thickness. The resistivity results show good trend correlation for almost for all VES. The ISM for VES locations showing a greater number of layers that there is larger variation in resistivity values for stations, VES_11, VES_14, VES_15, VES_22, VES_23 and VES_25.

4.1   Correlation of Subsurface Resistivity with Existing Borehole Data

There is a considerable overlap of resistivity values between different rock types. In addition, the resistivity of rocks depends on degree of weathering, fracturing, porosity, saturation, clay content and resistivity of pore water and the concentration of dissolved salts. The given rock type can exhibit a large range of resistivity. Identification of a rock type is difficult based on only resistivity data. Since the resistivity ranges of different earth materials have overlapping values, the calculated resistivity values cannot be directly applied to the soil or subsurface rock without correlating that with the subsurface formations. Correlation studies of resistivity and well data in nearby area is essential to identify the formation. The interpreted results of the VES are correlated with the available litholog of the exploratory boreholes to establish the correlation between resistivity and available lithological formations [11]. The interpreted geoelectrical parameters from both IR and ISM results are correlated with the nearest deepest borehole (Drill Hole -DH) as shown in Fig. 6.

The geoelectrical results obtained by the ISM are in good agreement with the available descriptive lithological logs (ground truths). The subsurface model derived by inverse slope method is more comparable with borehole data compared to conventional curve matching-based inversion method. However, the reasonable correlation was not observed for all the boreholes. The reasons may be due to insufficient depth of these boreholes and the considerable distance from the VES stations. The average
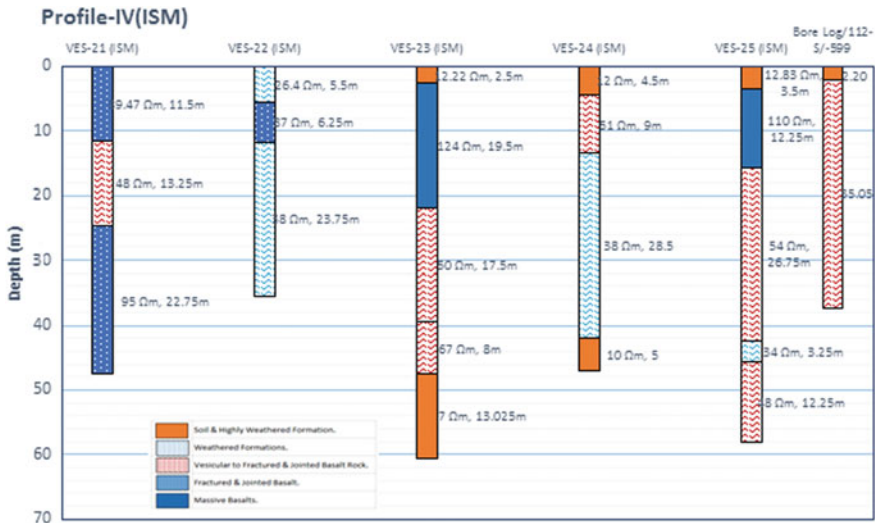
Fig. 6 Inferred geoelectric and geological sections, Profile-IV (ISM)

distance of the VES stations from boreholes were 285 m, 210 m, 200 m and 77 m for profiles I, II, III and IV, respectively. It is very likely that the geology may show changes over such considerable distance separation. Another possible reason may be difference of ground elevation between boreholes and VES locations due to topographic relief. The VES locations are on the downstream of the dam alignments and are on depressed floor elevation particularly for profile-I and profile-II. The difference between the average elevation of Profile-I, Profile-II and corresponding borehole location was more than 16.25 m and 8.5 m, respectively.

The ISM results show better correlations with borehole lithologs. Inverse slope method is found to be more suitable for identifying the deeply located formations. Hence, interpreted resistivity and thickness values by inverse slope method were adopted for layer geoelectric properties correlation. The obtained absolute resistivity from ISM was compared with known resistivity and associated lithology correlation for Deccan Trap formation [12]. Table 2 represents the correlation of absolute resistivity for the VES-25 with the drill hole DH-112S (CH-599 m) for identifying possible lithology.

4.2 Isopach and Isoresistivity Distribution

The resistivity values for the individual layer have been used to prepare isoresistivity maps showing distribution of electrical resistivity for each layer. The thickness isopach map for each layer is also constructed showing individual layer thickness

Table 2 Correlation of VES-25 with DH-112S/chainage CH-599 m

Borehole Data DH-112-S/-599			Geoelectric interpretation VES-25				
Formation	Tk. (m)	Depth up to (m)	No. of layer	Tk. (m)	Depth up to (m)	ρ (Ω m)	Formation
Overburden	2.20	2.20	1	3.5	3.5	12.83	Highly weathered
Chlorophaeitic compact porphyritic basalt with plagioclase phenocrysts	35.05	37.25	2	12.25	15.75	110	Jointed rock
			3	26.75	42.5	54	Vesicular, fracture and jointed rock
			4	3.25	45.75	34	Slightly weathered
			5	12.25	58	48	Vesicular, fracture jointed rock

variation. The layer thickness and layer resistivity contours are shown with colour variation for each layer (Figs. 7 and 8).

True resistivity values for resistivity inversion (IR) and inverse slope method (ISM). Both the methods have demonstrated high resistivity values for deeper layers. The resistivity profiles plotted with these high magnitude values for isoresistivity maps and resistivity profiles suppress and hinder the low-resistivity zones. Also, for the Deccan Trap, higher resistivity value above 300 Ωm indicates hard massive basaltic bedrock.

Considering these aspects, only for representation purpose (in Fig. 9), the high resistivity values are rationalized to 300 Ω m and infinite thickness of the deepest layer rounded to 100 m without losing their significance. These figures have helped to locate the low-resistivity formations in substratum. Figure 9 shows low-density formations around 30 m towards spillway section which may be a potential cause for seepage problem.

4.3 Depth to Competent Bedrock at VES Stations

The resistivity of formation depends upon the characteristic of the element constituting the given geologic medium. Rock resistivity is affected by the degree of fractures, weathering, porosity and saturation. The resistivity data obtained for both IR and ISM method was interpreted in terms of bedrock availability. The comparison of resistivity profiles showing variation of true resistivity with depth for both IR and ISM. The respective profiles show considerable qualitative agreement. The respective profiles depict the similar trend of high true resistivity contour closer and their approximate locations. However, there is quantitative difference for the absolute

Fig. 7 Stacked layer absolute resistivity contour map (ISM)

layer resistivity and thickness. The differences are mainly due to the high range of the resistivity values obtained for IR method.

The VES results revealed heterogeneous nature of the subsurface geological sequence. The geologic sequence beneath the study area is composed of highly weathered formation, slightly weathered soft rock formations, vesicular, fracture and jointed rock, fracture and jointed rock, thick jointed rock, massive basalt rock and highly massive basalt basement. It is also concluded from the study that the depth to the competent bedrock is relatively variable by both resistivity inversion and inverse slope method as shown in Fig. 10.

The depth to bedrock by IR method is in the range of 13–67 m and for ISM is about 30–58 m. The ISM giving approximately 20–21% higher average depth to the bedrock for left abutment and approximately 145% higher depth for right abutment. For the right abutment (VES_21 to VES_25), resistivity inversion method has given the less
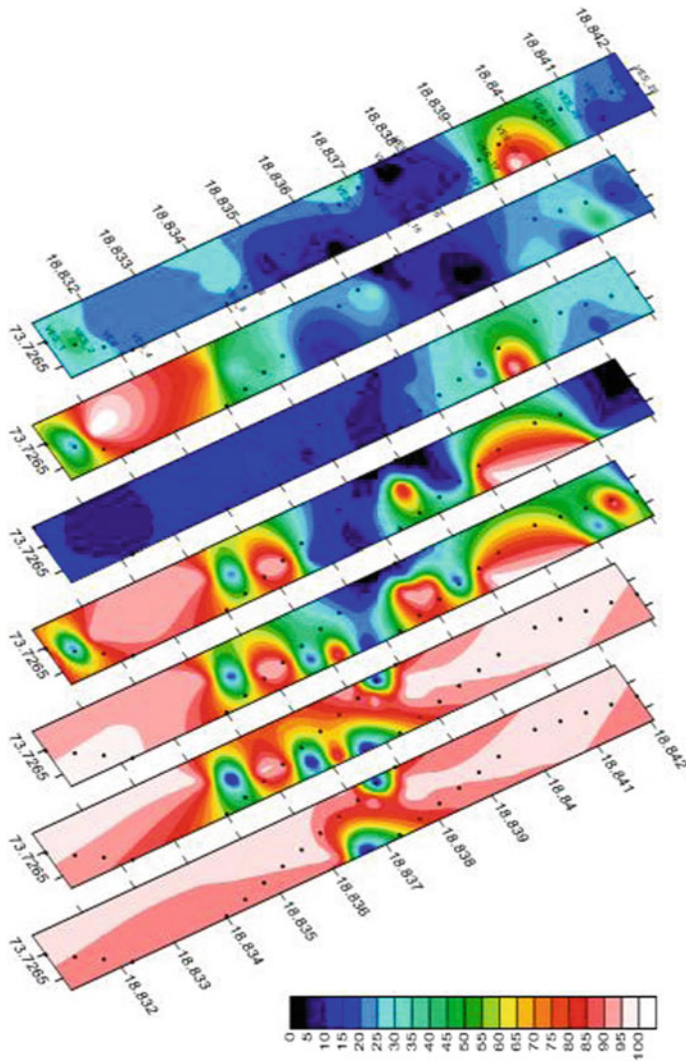
Fig. 8 Stacked layer thickness contour map (ISM)

depth to bedrock because of occurrence of thin high-resistivity layer at comparatively shallow depth (8.78 m, 2.17 m, 5.37 m, 2.03 m, 2.08 m, respectively). Thus, it has failed to consider the average resistivity of layers underneath this thin layer. The ISM decompresses the layers at greater depth identifying the better layer resistivity variation at a greater extent as compare to IR. Thus, giving a greater number of layers at greater depth. In IR method, the lower layers are compressed on logarithmic scale. Each geoelectric layer obtained by the investigations is not exactly the response from the individual litho units but the response of equivalent resistivity or the average of
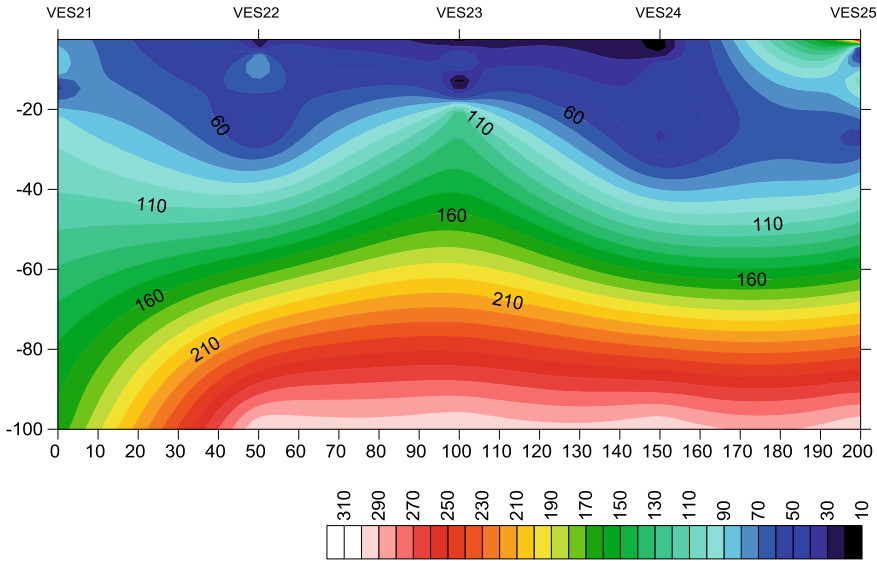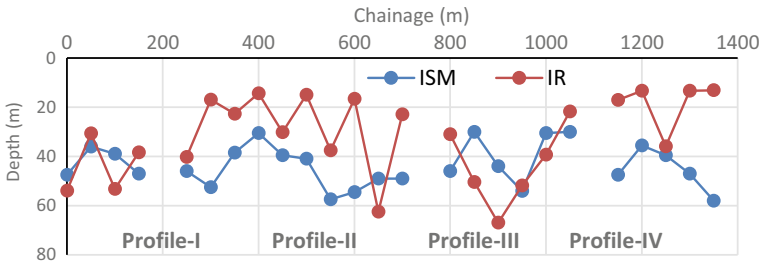
Fig. 9 Profile-IV VES 21_25 (ISM)

Fig. 10 Bedrock level comparison parallel to dam alignment

number of subsurface litho units. The resistivity variation observed in both methods is justified based on the different approaches of these methods.

It is also deduced that the study area can support low to high engineering structures as availability of competent bedrock normally around 30–40 m in most of the area underlain by fresh basement rocks at comparatively shallow depth. The study area can support engineering structures because of their shallow depths to the underlying rock after removal of the overburden soil and weathered formations.

The subsurface model derived by inverse slope method (ISM) is more comparable with borehole data compared to conventional curve matching-based resistivity inversion method (IR). Overall, the vertical electrical sounding (VES) interpretation by ISM have shown substantial correlation with the geological exploration data. (Fig. 6). The ISM is found to be more promising for interpretation of intermixed,

Table 3 Average depth to competent bedrock at VES points

Method	Depth to competent bedrock/foundation depth (m)		
	Left abutment (VES_1 to VES_7)	Middle portion (VES_8 to VES_20)	Right abutment (VES_21 to VES_25)
IR method	37	35	19
ISM method	43.8	42.7	45.5
Avg. foundation depth adopted	7.49		26.71

Table 4 Weaker zones in foundation strata at VES points

VES station	Approx. chainage (m)	Inverse slope method (ISM)		
		Location	Depth to bed rock (m)	Remark
VES_3	700	Left abutment	39.00	Bedrock followed by weaker section
VES_18	−42	Middle portion	54.00	Bedrock followed by weaker section
VES_23	−340	Right abutment	39.50	Fractured Bedrock followed by weaker section

composite, non-stratified igneous formations. Table 3 shows the calculated average depth to competent bedrock at the left abutment, middle portion and right abutment of dam.

The competent bedrock is available normally around 30– 40 m for most of the project area, which is underlain by fresh basement rocks. Resistivity survey has ascertained the occurrence of existing weaker zones identified by core logs, and it has also mapped the weaker zones in foundation strata at greater depth, which were unnoticed due to insufficient depth of conventional exploration. Also, Table 4 shows the weaker zones in foundation strata mapped using resistivity method. Towards the left dam abutment (Irrigation Cum Power Outlet—ICPO section), the resistivity increases with depth, implying less heterogeneity, thus indicating the consistent geological formation there. However, more resistivity anomalies were observed towards dam right abutment (spillway) section. These finding are in consistent with the known site geology [13]. The result interpretation suggests that the failure at the spillway section is likely due to seepage through these weaker zones (Figs. 3, 5 and 9). Therefore, it is recommended that these low-density sections need to be grouted to the appropriate depths.

VES results are also consistence with geological studies and Central Water and Power Research Station, Pune findings. Towards spillway location there is low-resistivity formation (35 Ω m) at depth of 40–45 m (Fig. 6) (Profile-IV) and Table 2. The results correlated with the geological data established that at the tail channel, there is an occurrence of red Tachylytic basalt with lava matrix, near to the stilling

basin area (energy dissipating assembly—EDA portion). The low-density Tachylytic basalt with lava matrix formation that is sandwich between fractured and jointed compacted basalt has seepage potential.

4.4   Limitations of Resistivity Methods

Resistivity interpretations may be ambiguous, thus, requiring independent conventional geological controls for verification. At dam valleys, the results of resistivity surveys are affected by the irregular terrain topography and by differences in electrical properties of dry material at the abutments and the wet material beneath the valley floor. The basic limitation with all 1-D resistivity inversion methods is that they only consider vertical variations in the subsurface resistivity. The electrical profiling (EP) is better than VES method for subsurface geological layers having lateral spatially variable resistivities [14].

5   Conclusion and Recommendations

The primary objective of the study was to enhance the geological understanding of site through resistivity study. A definitive conclusion cannot be drawn based on the resistivity study alone. The findings of resistivity survey complement available exploration data. After comparing the results, the following inferences can be made,

1. The basic purpose of investigation was to find depth of rock and major features like fracture at different depths and profile of hard rock. Location of anomalous foundation features and delineation of boundaries between overburden soil, weathered zones and fresh rock.
2. Electrical resistivity method when combined and correlated with results of the site exploration, improve overall geological understanding, subsequently reducing risk in the overall analysis. The resistivity interpretations are consistent with known site geology. The interpretation has ascertained the weak zones identified in geological investigation. Hence, the resistivity method can be used to identify problematic areas that needs careful consideration.
3. The electrical resistivity investigation has revealed the inadequacy of depth of geological exploration at study site. Electric resistivity survey prior to core drilling may have helped to plan the further detailed investigation and proper site selection for spillway location. Dissimilarities between profiles and boring log data can exist due to the heterogeneity of the site, as well as the resolution capabilities and data smoothing associated with each method.

References

1. Turrini C et al (2014) Present day 3D structural model of the Po Valley basin, Northern Italy. Mar Pet Geol 56:266–289
2. Kulkarni SR (2006) Engineering geological report of spillway and chute channel of Bhama Askhed project. Geological Consultancy Report, Pune
3. Jatau BS et al (2013) The use of vertical electrical sounding (VES) for subsurface geophysical investigation around Bomo area, Kaduna state, Nigeria.IOSR J Eng 3 (2013). https://doi.org/10.9790/3021-03141015
4. Loke MH (2000) Electrical imaging surveys for environmental and engineering studies. A practical guide to 2-D and 3-D surveys
5. Narayan S, Ramanujacharya KR (1967) Short note—an inverse slope method of determining absolute resistivity. Geophysics XXXII(60):L1036–1040
6. Bobachev C (2002) IPI2Win—a windows software for an automatic interpretation of resistivity sounding data. Ph. D, Moscow State University, Moscow
7. Waswa AK (2019) Application of electrical resistivity method in mapping underground river channels: a case study of Kabatini Area in the Kenyan Rift Valley. Univer J Geosci 7(1):1–14. https://doi.org/10.13189/ujg.2019.070101
8. Ramanujachary KR (2012) A monograph on geophysical techniques for groundwater exploration with special reference to resistivity techniques. In: Integrated geo instruments & services. Professional Book Publisher, Hyderabad . ISBN 978-81-909728-2-6
9. Usama M et al (2015) 1D and 3D inversion of VES data to outline a fresh water zone floating over saline water body at the north western coast of Egypt. National Research Institute of Astronomy and Geophysics (NRIAG), Egypt
10. Nejad HT et al (2011) Vertical electrical sounding (VES) resistivity survey technique to explore groundwater in an arid region, Southeast Iran. J Appl Sci 11:3765–3774. https://doi.org/10.3923/jas.2011.3765.3774
11. Kulkarni SR (1994) Engineering geological studies at Bhama irrigation project. Taluka Khed. Geological Consultancy Report, Pune
12. Erram VC et al (2012) Hydrogeology of Deccan traps and associated formations in peninsular India. Memoir Geol Soc India 80, 51–66
13. Aswar DS et al (2018) Geomodelling for Dam Foundation Stratum. http://hdl.handle.net/10603/265972
14. Salman AM et al (2020) Application of the electrical resistivity method for site investigation in university of Anbar. Iraqi J Sci 61(6), 1345–1352

An Extensive Study of SegNet Model in Automatic Brain Tumor Segmentation Using Multi-modal MR Scans

Suchismita Das, G. K. Nayak, and Sanjay Saxena

Abstract Glioma is one of the most dangerous types of brain tumor due to its aggressive and fast-growing nature. Patients suffer from high grade survive very less time as compare to other tumors. The main aim of this article is the analysis of the SegNet performance on brain tumor segmentation in comparison to other encoder–decoder models. SegNet model is implemented through its pooling indices which takes less memory as compare to other semantic segmentation models. Also, the preprocessing of medical images is performed to improve the accuracy along with multi-parametric MRI scans. The experiment was computed on BraTs 2017, and the model achieves dice score of 0.91 for complete tumor with FLAIR, core 0.92, and 0.86 for core and enhancing, respectively, with $T1$ce modality and 0.78 for edema with $T2$-weighted MRI scan, and it outperforms the state-of-art.

Keywords MRIs · SegNet · Deep learning · AUC · Brain tumor

1 Introduction

Medical imaging and correct lesion segmentation from other brain artifacts in MRI images of the brain will aid in giving an error-free diagnosis. Lesions have been a cause of massive structural deformations that have often led to incorrect segmentation. It is very difficult task of evaluating and extracting correct information from pathologic seimen due to their subjective and complex nature. From the statistical analysis, 85–90% of all primary central nervous system (CNS) tumors [1] are accounted by brain tumor. Malignant tumors are divided into two types: primary tumors originated within the brain, and secondary tumors, spread to brain, originated from other organs of the body. When it comes to glioblastoma, it is just a mere 7.2%. The median survival has been observed to be only eight months. Since the tumors

S. Das (✉) · G. K. Nayak · S. Saxena
CSE Department, International Institute of Information Technology, Bhubaneswar, Odisha, India
e-mail: Suchismita.dasfcs@kiit.ac.in

S. Das
CSE Department, KIIT Deemed to be University, Bhubaneswar, Odisha, India

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
S. K. Udgata et al. (eds.), *Intelligent Systems*, Lecture Notes in Networks and Systems 431, https://doi.org/10.1007/978-981-19-0901-6_33

advance rapidly, it is stated that glioblastoma is the most aggressive and takes lives quickly. Glioma, which develops from glial cells, is one of the most common malignant tumor types [2]. It is categorized into low-grade glioma (LGG), which grows slowly, and highly malignant tumors known as high-grade glioma (HGG), which can be life-threatening. HGG is untreatable in some cases, even with advanced imaging, radiotherapy, and surgical techniques. HGG segmentation is a challenging task [3, 4] due to cell mutations, irregular growth in multiple tissues, and the complex and diverse nature of HGG. Treatment is also challenging because tumor characteristics vary from patient to patient; therefore, it can be difficult to use a large number of scans and volumes when attempting to visualize and quantify lesions.

High-grade gliomas are still untreatable [2] in few cases even having advanced imaging, radiotherapy, and surgical methodology. In different imaging techniques, magnetic resonance imaging (MRI) characteristics are considered one of the preferred ways to diagnose [5, 6] a brain tumor as it creates a detailed and clear picture than CT scans. Different contrasts images produced through multimodal MRI protocols provide complementary information which helps in segmenting the brain tumor and its surrounding tissues that are playing important role in diagnosis and treatment of brain tumor. There are mainly three regions in brain tumor such as necrotic and non-enhancing tumor, peritumoral edema, and enhancing tumor. The most common MRI sequences are native or $T1$-weighted, post-contrast $T1$-weighted ($T1$ce), $T2$-weighted and fluid-attenuated inversion recovery (FLAIR) volumes. Each one of these above modalities has a different contrast which helps in segmenting [6] the sub-tumors through their characteristics.

Routine image acquisitions throughout the course of a patient's lesion treatment can overwhelmingly help medical facilitator's in the best diagnosis, planning, and treatment of a patient. There are different methods proposed for automatic segmentation of brain tumor using traditional methods as well as machine learning [7, 8] approaches. As described in [9], neural network, SVM, and clustering algorithms outperform than the traditional methods such as threshold based, region growing, and watershed. But, machine learning algorithms are based on selecting the feature vector which influenced the performance in a greater way. To overcome this, now a days, deep learning is achieving a great success and researcher's attention in medical image segmentation [10, 11] in which the deep features of the images are extracted automatically through a series of convolutional operations; commonly used CNN models are modified in different ways to make it flexible for image segmentation operations.

The U-Net has been widely used architecture [12, 13] in medical image analysis. The model considers the full feature map during upsampling which makes the network larger in terms of memory. To make the model memory efficient, the next deep model known as SegNet encoder–decoder model is explored. In this work, an automated deep learning-based brain tumor segmentation technique by well-known SegNet encoding–decoding model [14–16] was implemented.

2   Literature Survey

Automatic brain tumor segmentation is a complex and challenging task due to the aggressive and variability characteristics of the tumor. Brain tumor segmentation is through three generation. There is a numerous research going on cancer detection by using three generation: traditional methods, machine learning, and deep learning techniques [17, 18]. Here, some literature based on the brain tumor segmentation using deep learning is discussed briefly.

Currently, deep learning convolution neural network is attracting the attention in medical domain due to its automatic extraction of feature from the given input image. It performs a set of convolutional operations on the input image and finally produces a feature vector which is responsible to characterize the given image in different fields of research. Semantic segmentation model SegNet is popularly used by many researchers due to its memory-efficient algorithm [2, 15]. Hao Dong et al. [2] proposed deep learning-based U-Net model which is also an encoder–decoder model but takes more parameters and memory due to the skip connection operation. They achieved a good DSC for core and complete, but enhancing result is not so good. Similarly, Salma Alqazzaz [14] developed a model to automatic segment brain tumor by focusing the most effective features and information extraction from different MRI scans using 3D dataset. Then, the MRI voxels are classified into tumor and sub-tumor parts using SegNet model followed by a decision tree (DT). It takes nearly about 3 days on a single NVIDIA GPU Titan XP and gives better accuracy for core and enhanced region as compare to the state-of-the-art.

3   Methodology

The main aim of this study is to analyze the SegNet architecture in terms of brain tumor segmentation. This is achieved in three main steps: a data preprocessing step to remove biases and do the normalization of the given MRI dataset, a training step to train the SegNet for all the modalities, and finally computing the performance in terms of well-known segmentation metrics.

In this study, high-grade glioma MRI dataset was considered, as low-grade glioma patients are larger survival rate as compare to HGG. The most commonly used standard BraTS 2017 and BraTs 2018 dataset were taken for evaluating the results. Preprocessing is carried out to minimize the different intensity inhomogeneity, artifact such as scanner variability, patient motions. For noise removal, median filter is applied to all 3D MRI modalities for each patient. The unwanted artifacts are removed through N4ITK bias field correction [18, 19] to get better accuracy. Intensity normalization is performed and then clipping the images automatically to smaller size. Due to the suppression of background and focusing on only tumor portion, training time and memory requirements are decreased along with the class imbalance problem.

This model is a typical SegNet model [20] which is similar to the encoding–decoding model depicted in Fig. 1. It consists of five encoding blocks and five decoding blocks. The first two encoding blocks have two conv. operational layers, and the next three encoding blocks contain three conv. operational layers. Instead of the max pooling layer, a custom layer termed as Max-Pooling-WithArgMax is used which performs typical max pooling on the feature maps with stride of 2×2 and kernel size of 2×2 but also stores the index position of the max pooling pixels. Instead of the upsampling layer, a custom layer termed as Max-UpSampling-WithArgMax is used which uses the corresponding index values stored during encoding to upsample the feature maps. The final layer of this model is a convolutional layer of kernel size 1×1 and number of feature maps equal to the number of classes and is followed by a sigmoid activation function to get a probability feature map. In our methodology, ReLU is used in intermediate convolutional block, and sigmoid is used in output layer to do classification. For faster convergence and enhanced performance, the combined binary cross-entropy and dice loss function are taken as BCE dice loss for our segmentation task.
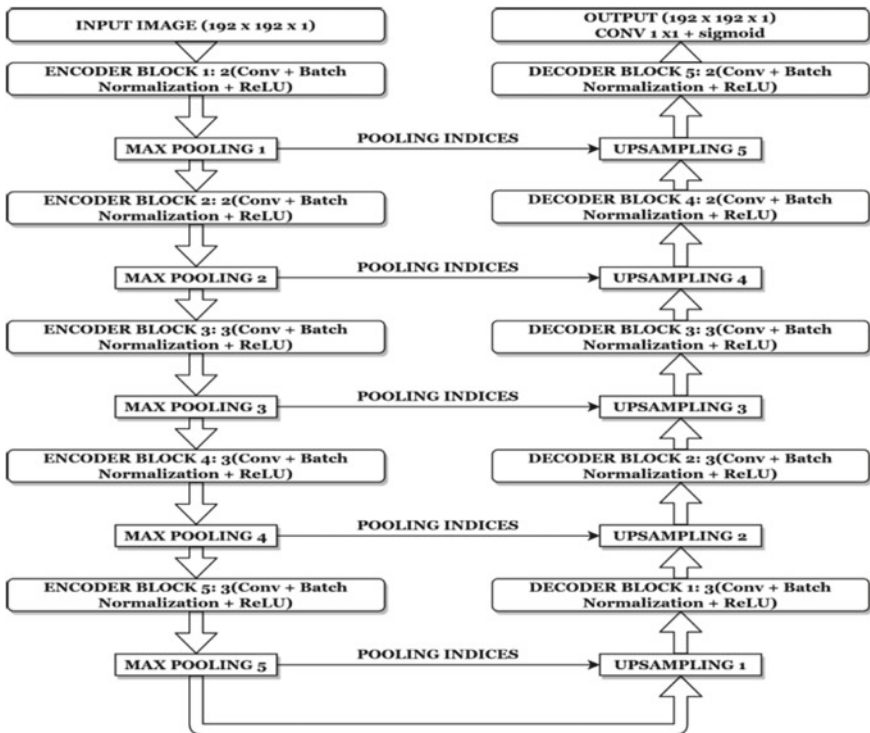

Fig. 1 SegNet architecture used in our proposed work

4   Results and Discussions

In this section, we are going to discuss regarding the dataset used for evaluation, different model parameters, training details, and segmented results.

4.1   Implementation Details

The SegNet model performance has been examined and evaluated using the BraTS 2017 and BraTs 2018 data [9]. There are total 210 high-grade glioma and 75 low-grade glioma patient subjects are available in BraTS 2017 dataset. There are total 210 high-grade glioma and 75 low-grade glioma patient subjects are available in BraTS 2017 dataset. Wherever, this study covers only 210 HGG patient dataset counted by us. Due to low risk and high survive rate of LGG patients as compare to those with HGG [21], these subjects are not included for the evaluation. The input image slice is trimmed automatically to 192 × 192 after gone through data preprocessing steps. The whole dataset of 210 patients is splitted into training and testing as 8:2 ratio. The training image dataset is again divided into training and validation data randomly. To avoid the over fitting due to less number of images, data augmentation is applied to the whole training dataset. To achieve an effective and efficient training, the mini batch size was set to 32 image slices, and fixed number of epochs equals to 100. In model, the Adam optimization [22, 23] is used with default learning rate of 0.001. The Adam algorithm is an adaptive learning rates [24] methods which computes individual learning rates for each parameter. Several evaluation metrics were used to evaluate the segmented tumor sub-regions with the segmented ground truth. Dice score compares the overlays between the predicted and the ground truth. Other metrics include accuracy and IoU which are the standards for classification tasks are used to get deeper insights about the predictions.

4.2   Experimental Results

The network structure and the number of trainable and non-trainable parameters of the model are shown in Table 1. In SegNet, a total of 24,957,057 parameters are used; out of which, 24,942,721 are trainable params, and 14,336 are non-trainable params. While training through SegNet, its param requires 2.9 GB memory for a batch size of 32 images which is memory efficient as compare to other encoder–decoder model. In contrast to memory, SegNet model takes average16.5 h. training time for each of the modalities. Whereas, SegNet stores only the pooling indices of feature map during max pooling in turn requires less memory than other model. But, through the experiment, it is observed that it takes higher computation time to reconstruct the feature map from pooling indices in decoding phase.

Table 1 Detailed structure of SegNet and encoder–decoder model

Layer type	Output size	Filter size	No. of parameters (param #, total = 24,957,057 Trainable params: 24,942,721 Non-trainable params: 14,336)
Input layer	$192 \times 192 \times 1$	–	0
2 @ (Conv. + BN)	$192 \times 192 \times 64$	3×3	38,080
max_pooling	$96 \times 96 \times 64$	2×2	0
2 @ (Conv. + BN)	$96 \times 96 \times 128$	3×3	222,464
max_pooling	$48 \times 48 \times 128$	2×2	0
3 @ (Conv. + BN)	$48 \times 48 \times 256$	3×3	1,478,400
max_pooling	$24 \times 24 \times 256$	2×2	0
3 @ (Conv. + BN)	$24 \times 24 \times 512$	3×3	5,905,920
max_pooling	$12 \times 12 \times 512$	2×2	0
3 @ (Conv. + BN)	$12 \times 12 \times 512$	3×3	7,085,568
max_pooling	$6 \times 6 \times 512$	2×2	0
Upsampling layer	$6 \times 6 \times 512$	2×2	0
3 @ (Conv. + BN)	$12 \times 12 \times 512$	3×3	7,085,568
Upsampling layer	$24 \times 24 \times 512$	2×2	0
3 @ (Conv. + BN)	$24 \times 24 \times 256$	3×3	1,832,064
Upsampling layer	$48 \times 48 \times 256$	2×2	0
3 @ (Conv. + BN)	$48 \times 48 \times 128$	3×3	591,744
Upsampling layer	$96 \times 96 \times 128$	2×2	0
2 @ (Conv. + BN)	$96 \times 96 \times 64$	3×3	111,232
Upsampling layer	$192 \times 192 \text{x} 64$	2×2	0
2 @ (Conv. + BN)	$192 \times 192 \times 64$	3×3	74,368
Output layer	$192 \times 192 \times 1$	3×3	577

The performance of the segmentation is evaluated through most commonly and useful metrics [25] for image segmentation: sensitivity, specificity, dice score, and accuracy measurements. The detailed metric for each of the model is shown in Table 2. The above metrics calculate the efficiency of the algorithm regarding correctly classifying a pixel if it belongs to tumor region. Next, we have considered the error of prediction, i.e., the analysis is also carried out by measuring the mean squared error of the test dataset. In statistics, the mean squared error (MSE) of model measures the average squared difference. SegNet is having MSE value of 0.021142694 between the predicted and actual output. The segmentation results are demonstrated in Fig. 2 on different test images. Figure 2 depicted some semantic segmentation results in visual form using SegNet model.

The SegNet model performance is also demonstrated through the different graphs representing accuracy, dice, and IOU, versus number for iteration during training.

Table 2 Performance for each of sub-tumor segmentation, #Acc: Accuracy, #Spe.: Specificity, # Sen: sensitivity

	Complete			Core			Enhancing			Edema		
	Sen	Spe	Acc	Sen	Spe	Acc	Sen	Spe	Acc	Sen	Spe	Acc
FLAIR	0.98	0.94	0.98	0.98	0.79	0.98	0.98	0.54	0.97	0.98	0.68	0.97
T1-weight	0.98	0.85	0.97	0.98	0.80	0.98	0.84	0.47	0.70	0.98	0.57	0.96
T1ce	0.98	0.87	0.97	0.99	0.93	0.99	0.99	0.89	0.99	0.98	0.69	0.97
T2-weight	0.99	0.91	0.98	0.99	0.86	0.98	0.99	0.61	0.98	0.98	0.68	0.97
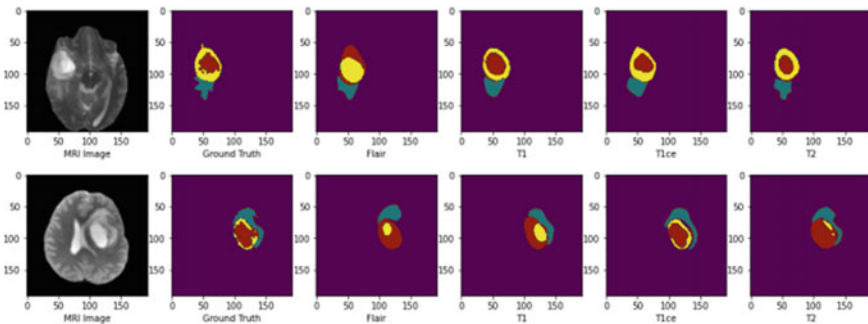
Fig. 2 Segmentation using SegNet model. From leftmost, images are as follows: MRI scans, ground truth, tumor-flair, tumor-T1, tumor-T1ce, tumor-T2, respectively

Dice score is considered as one of the most common metric to evaluate the performance of the segmentation algorithm. The model is evaluated here by using validation dice score which considered to be maximized during training instead of mean squared error. Dice score is measured for individual image and then computed as overall dice score for a set of validation data as shown in Fig. 3a. Similarly, Fig. 3b–d shows the accuracy, IoU, and loss of the model during segmentation of the interested region during training.

The research also demonstrates the performance in terms of receiver operating characteristic (ROC) curve taken between false positive rate vs. true positive rate and measures the area under the curve (AUC value) to analyze the performance of classification of pixel in details. The AUC can be used to compare the performance of two or more classifiers in absolute term: higher the AUC better the segmentation/classification. The details of AUC value for each of the sub-tumor segmentation are shown graphically in Fig. 4. There is a similar trend in AUC values during segmenting the sub-tumor region sing FLAIR, T1-weighted, T-weighted, and T1ce modalities. For complete and edema region exaction, FLAIR and T2-weighted are performing better as compare to T1-weighted and T1ce. Similarly, for core and enhancing, the performance is better in T1ce as compare to other three modalities.
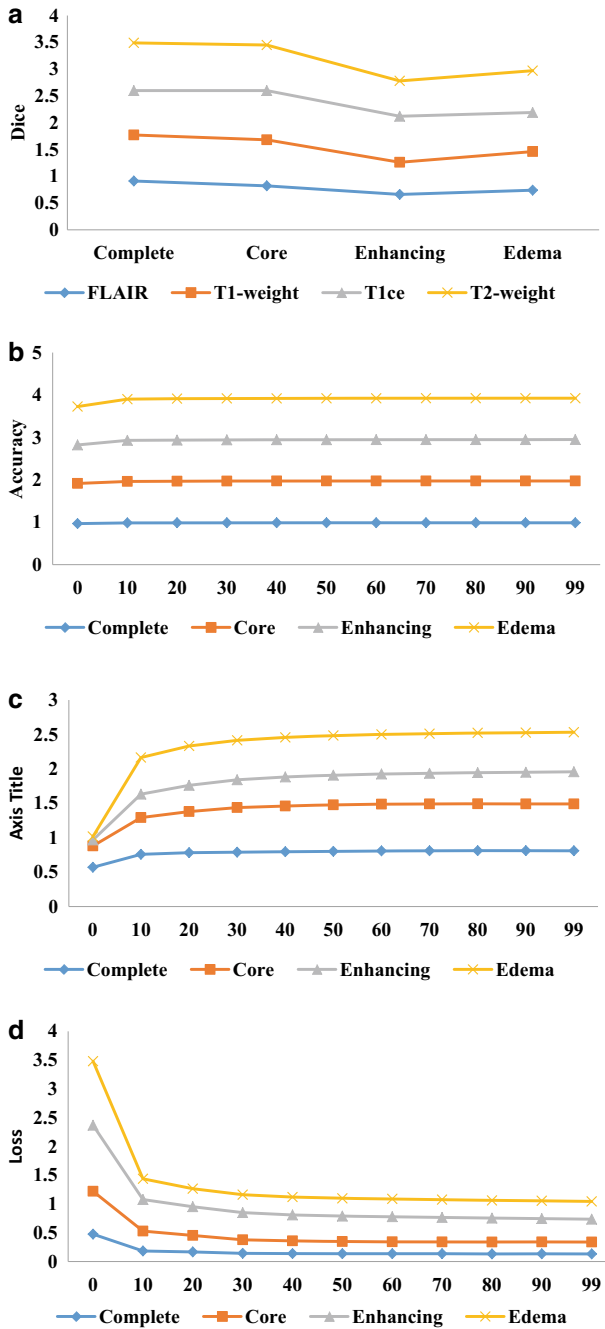
Fig. 3 **a** Dice score versus iteration for sub-tumor segmentation. **b** Accuracy versus iteration for sub-tumor segmentation. **c** IoU versus iteration for sub-tumor segmentation. **d** Loss versus iteration for sub-tumor segmentation
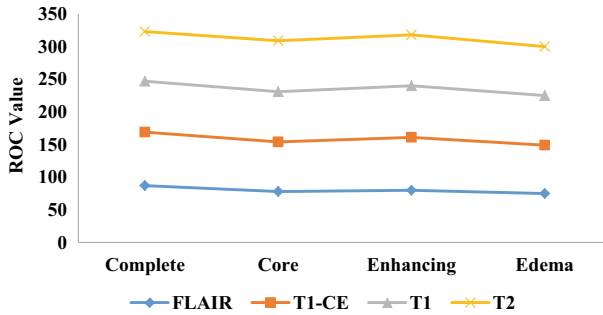
Fig. 4 AUC/ROC value for complete, core, enhancing, and edema sub-tumor segmentation

The result also shows the model efficiency in terms of ROC curve (AUC) for sub-tumor part with respect to the different modalities. The comparison of the proposed model with other deep learning model is shown in Table 3.

5 Conclusion and Future Scope

In this paper, a SegNet model is analyzed in details in the field of brain tumor segmentation. It efficiently and automatically segment a hole tumor along with its sub-tumor parts. To increase the efficiency of the model, we have used the thresholding in selection of image slices for training and validation. Our results and analysis show that our method performs well in segmenting each of the sub-tumors. Due to the pooling indices used in SegNet, the model becomes memory efficient but takes more computational time. The edema region is also extracted which is not computed in most of the research articles and got a satisfactory result in comparison to state-of-the-art. Along with the model, we compare the different MRI modalities for extracting different sub-tumor region. Post-contrast $T1$-weighted ($T1$ce) MRI scan outperforms other three in segmenting the core and enhancing tumor region, while $T2$-weighted is giving highest accuracy in extracting whole tumor and edema region. The research can be further extended with performance analysis of SegNet with other encoding models and can be ensemble to design a memory and computational efficient hybrid model to get better accuracy.

Table 3 Comparison of our method and other methods

Author	Year	Method	Metric	Dataset	Complete	Core	Enhancing	Edema
Yi [26]	2016	3D fully CNN	DSC	BRATS 2015 training (HGG + LGG)	0.89	0.76	0.80	–
Dong [2]	2017	U-Net	DSC	BRATS 2015 training datasets (HGG)	0.88	0.87	0.81	–
Kamnitsas [27]	2017	3D CNN	DSC	BRATS 2015 training (HGG + LGG)	0.90	0.76	0.73	–
Xu [13]	2019	LSTM + U-Net	DSC	BRATS 2015 training (HGG + LGG)	0.73	0.62	0.42	–
Alqazzaz [14]	2019	SegNet + RF	F1-score	BRATS 2017 training (HGG + LGG)	0.85	0.81	0.79	–
Lyksborg [28]	2015	Ensemble model	DSC	BRATS 2014 training	0.81	0.697	0.681	–
Feng [29]	2020	Ensemble model	DSC	BRATS 2018 training (HGG + LGG)	0.9114	0.8304	0.7946	–
Our model	2021	SegNet	DSC	BRATS 2017 training (HGG)	0.85	**0.91**	**0.88**	**0.83**
			F1-score		**0.92**	**0.95**	**0.93**	**0.84**

The result of our proposed method is given in bold to specify it clearly

References

1. Farmanfarma KK, Mohammadian M, Shahabinia Z, Hassanipour S, Salehiniya H (2019) Brain cancer in the world: an epidemiological review. World Cancer Res J 6(5)
2. Das S, Nayak GK, Saba L, Kalra M, Suri JS, Saxena S (2022) An artificial intelligence framework and its bias for brain tumor segmentation: A narrative review. Comput Biol Med, 2022 Feb 19,105273

3. Wadhwa A, Bhardwaj A, Singh Verma V (2019) A review on brain tumor segmentation of MRI images. Magn Reson Imaging 61:247–259
4. Angulakshmi M, Deepa M (2021) A Review on Deep Learning Architecture and Methods for MRI Brain Tumour Segmentation. Curr Med Imaging 17(6):695–706
5. Das S, Bose S, Nayak G, Satpathy S, Saxena S (2021) Brain tumor segmentation and overall survival period prediction in glioblastoma multiforme using radiomic features. Concurr Comput Pract Exp e6501
6. Das S, Nayak G, Saxena S, Satpathy S (2021) Effect of learning parameters on the performance of U-Net model in segmentation of brain tumor. Multimedia Tools Appl 1–19
7. Duraisamy S, Muthukumaravel A. Automated frame work for brain tumor segmentation using supervised learning method
8. Huang L, Ruan S, Denoeux T (2021) Belief function-based semi-supervised learning for brain tumor segmentation. In: 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI), pp 160–164. IEEE
9. Menze B, Jakab A, Bauer S, Kalpathy-Cramer J, Farahani K, Kirby J, Burren Y, Porz N, Slotboom J, Wiest R, Lanczi L (2015) The multimodal brain tumor image segmentation benchmark (BRATS). IEEE Trans Med Imaging 34(10):1993–2024
10. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, vol 25, pp 1097–1105
11. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 580–587
12. 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, pp 234–241
13. Xu F, Ma H, Sun J, Wu R, Liu X, Kong Y (2019) LSTM multi-modal UNet for brain Tumor segmentation. In: 2019 IEEE 4th international conference on image, vision and computing (ICIVC), 2019, pp 236–240. IEEE
14. Alqazzaz S, Sun X, Yang X, Nokes L (2019) Automated brain tumor segmentation on multimodal MR image using SegNet. Comput Vis Med 5(2):209–219
15. Padalkar GR, Khambete MB (2018) Analysis of basic-SegNet architecture with variations in training options. In: International conference on intelligent systems design and applications, 2018. Springer, pp 727–735
16. El Adoui M, Mahmoudi SA, Larhmam MA, Benjelloun M (2019) MRI breast tumor segmentation using different encoder and decoder CNN architectures. Computers 8(3):52
17. Kaldera H, Gunasekara S, Dissanayake MB (2019) MRI based glioma segmentation using Deep Learning algorithms. In: 2019 International research conference on smart computing and systems engineering (SCSE), pp 51–56. IEEE
18. Cong W, Song J, Luan K, Liang H, Wang L, Ma X, Li J (2016) A modified brain MR image segmentation and bias field estimation model based on local and global information. Computat Math Methods Med 2016
19. Song S, Zheng Y, He Y (2017) A review of methods for bias correction in medical images. Biomed Eng Rev 1(1)
20. Badrinarayanan V, Kendall A, Cipolla R (2017) Segnet: a deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans Pattern Anal Mach Intell 39(12):2481–2495
21. Banerjee S, Mitra S, Masulli F, Rovetta S (2019) Deep radiomics for brain tumor detection and classification from multi-sequence MRI. arXiv preprint arXiv:1903.09240
22. Xue C, Zhang J, Xing J, Lei Y, Sun Y (2019) Research on edge detection operator of a convolutional neural network. In: 2019 IEEE 8th Joint international information technology and artificial intelligence conference (ITAIC), 2019. IEEE, pp 49–53
23. Huang CL, Shih YC, Lai CM, Chung VYY, Zhu WB, Yeh WC, He X (2019) Optimization of a convolutional neural network using a hybrid algorithm. In: 2019 International joint conference on neural networks (IJCNN). IEEE, pp 1–8

24. Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980
25. Saxena S, Garg A, Mohapatra P (2019) Advanced approaches for medical image segmentation. In Application of biomedical engineering in neuroscience. Springer, Berlin, pp 153–172
26. Yi D, Zhou M, Chen Z, Gevaert O (2016) 3-D convolutional neural networks for glioblastoma segmentation. arXiv preprint arXiv:1611.04534
27. Kamnitsas K, Ledig C, Newcombe VF, Simpson JP, Kane AD, Menon DK, Rueckert D, Glocker B (2017) Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. Med Image Anal 36:61–78
28. Lyksborg M, Puonti O, Agn M, Larsen R (2015) An ensemble of 2D convolutional neural networks for tumor segmentation. In: Scandinavian conference on image analysis, 2015. Springer, pp 201–211
29. Feng X, Tustison NJ, Patel SH, Meyer CH (2020) Brain tumor segmentation using an ensemble of 3D U-nets and overall survival prediction using radiomic features. Front Comput Neurosci 14:25

A Parallelized Approach Toward Solving the Weighted Consensus Model for Classifying COVID-19 Infection

Nitin Sai Bommi and **Sarath Kumar Bommi**

Abstract CT scans are proved to be one of the best ways to identify the presence of COVID-19 and diagnose the patients. To reduce the workload on doctors, many researchers have come up with automatic classification techniques. However, all the research was done to improve the proposed model's accuracy. To improve the reliability and robustness of predictions, recent work [1] focused on allotting weightage to predictions. This paper aims at improving the performance of the existing one in both reliability and computational terms. We compare different metrics' ability to compute the weights of the base models and show the computational time improvement by parallelism. The proposed approach achieved about 4.2 speedup and an efficiency of around 60%. The proposed parallelizable approach works best when a bulk of test samples are to be tested to reduce the total testing time.

Keywords Machine learning · Weighted consensus model · Parallel computing · COVID-19

1 Introduction

The novel coronavirus (COVID-19) disease has been labeled a pandemic by the World Health Organization (WHO), raising public health concerns worldwide. COVID-19 is a widely distributed and highly infectious virus transferred from infected people through direct contact and indirectly through the air, surface and surroundings with which infected people come into contact. Fever, dry cough, headache, weariness, loss of taste and smell, and dyspnea are all symptoms of viral pneumonia in the lungs. The fact that the majority of those infected have asymptotic symptoms exacerbates the transmission of COVID-19. As a result, rapidly detecting the sick person's symptoms is essential. The pandemic scenario is touching billions of people on a social,

N. S. Bommi (✉)
University of Hyderabad, Hyderabad, India
e-mail: nitinsai.bommi@gmail.com

S. K. Bommi
Indian Institute of Technology Jodhpur, Jodhpur, India

economic, and medical level. It is causing profound changes in social interactions and educational environments and affecting many people's lives.

Advanced technology is being sought by the healthcare industry to monitor, detect, and diagnose illness and quickly stop the spread of the COVID-19 pandemic. The two most common types of diagnosis in today's medical practice are RT-PCR and CT scans. Even though the vaccination is available, detecting the sickness early and immediately quarantining the person is critical to preventing the disease from spreading. The RT-PCR's sensitivity cannot always be overlooked. As a result, chest CT scans and X-ray images are used as a substitute technique to diagnose COVID-19 infection because of their great sensitivity.

The objective of the paper is to improve the robustness of the existing model [1] and to improve the testing time by parallelizing computations. The paper presents a range of metrics and uses them to calculate the weightage. The computations are also made quickly by using the idea of matrix multiplication. The paper is organized in the following manner: Sect. 2 describes the previous work and also the limitations, Sect. 3 elaborates the proposed methodology, Sect. 4 displays the results and Sect. 5 gives the conclusion and future scope.

2 Literature Survey

The COVID-Net model [2] was created to detect COVID-19 positive cases from chest radiography pictures, and it has a sensitivity of 80%. They also introduced COVIDx, an open-access benchmark dataset comprised of 13,975 CXR images across 13,870 patient cases from five open-access data repositories. Wang et al. [3] analyzed the radiographic variations in infected patients using CT scans. They created a model with an accuracy of 89.5% and employed the modified inception transfer learning technique. Further, the diagnosis is made by the extracted features.

Narin et al. [4] used chest X-ray images to diagnose COVID-19 using DCNN-based transfer models. They used the Inception-ResNetV2, InceptionV3, and ResNet50 models for accurate prediction. They employed three binary datasets: X-ray images of normal (healthy), COVID-19, bacterial, and viral pneumonia patients. Zhang et al. [5] also developed a novel seven-layered CNN-based creative diagnosis model that detects COVID-19 in chest CT scans with a sensitivity of 94.44%, a specificity of 93.63%, and accuracy of 94.03%.

Hussain et al. [6] proposed the use of plain chest X-ray and CT scan data to train a new CNN model called CoroDet for automatic detection of COVID-19. CoroDet was created to be an accurate diagnostic tool for two-class classification (COVID and Normal), three-class classification (COVID, Normal, and non-COVID pneumonia), and four-class classification (COVID, Normal, non-COVID viral pneumonia, and non-COVID bacterial pneumonia). They used 1 class, 2 class, and 3 class classifiers to attain accuracies of 99.1%, 94.2%, and 91.2%, respectively.

Zhang et al. [7] proposed a new method that combines DenseNet and the optimization of transfer learning setting (OTLS) strategy. Different learning factors were

assigned to three layers: frozen layers, middle layers, and new layers using a composite learning factor (CLF). The proposed approach had sensitivity, specificity, precision, and accuracy of 96.35 ± 1.07, 96.25 ± 1.16, 96.29 ± 1.11, and 96.30 ± 0.56, respectively.

2.1 Contributions of the Present Work

This paper is an extension to the base paper [1]. The contributions to the base paper and the previous works are listed below:

– The paper uses different metrics to calculate the weights.
– The paper proposes a parallel computation approach to speed up the execution time.

3 Methodology

The base paper [1] used basic Machine Learning models. We used the same base models and a dummy model to strengthen our results. The models are individually trained on the data and the weightage of each model is considered during the final prediction. The weightage is calculated by normalizing the accuracies of the base models on the test data. When working with imbalanced datasets, different assessment measures are frequently required. Unbalanced classification problems often evaluate classification errors with the minority class as more important than those with the majority class, in contrast to traditional evaluation metrics that deem all classes equally relevant. As a result, performance criteria that focus on the minority class may be required, which is difficult because the minority class is where we lack the necessary data to train an effective model [8, 9].

This paper presented a range of metrics during normalization and compared them. The base models are used to calculate different metrics and the values are normalized as shown below.
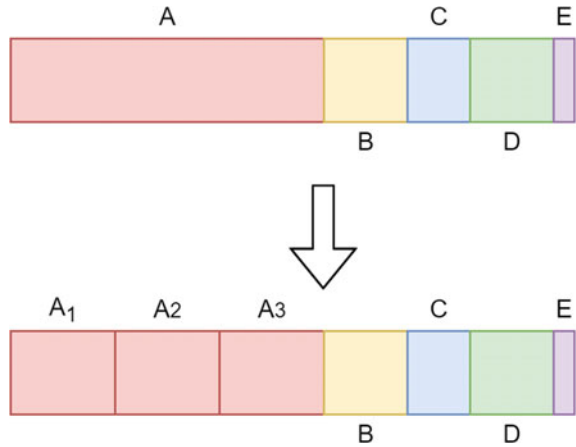
model_i denotes the metric value for the ith model

$$\text{total\_value} = \sum_{i=1}^{k} \text{model}_i$$

$$\text{weight}_i = \frac{\text{model}_i}{\text{total\_value}}$$

After the weights of different models for a metric were calculated, the base paper used a sequential approach to calculate the base predictions and weights. We propose

Fig. 1 Proposed parallelized approach for base predictions

a new parallelized approach toward solving the problem. Since the base models use scikit-learn[1] library throughout the experiment, the base predictions cannot be parallelized.

However, the test data can pass through the five base models in parallel by using multi-processing. This will allow us to compute the base predictions for each model on a different processor. The time taken for these base predictions is affected by the process taking the longest time to predict the test data. We found that the longest process takes more than thrice the next longest model. So, to improve the time, we divided the test set into three equal parts and ran the longest model to predict each test part on three different processors. Figure 1 shows how the longest process is distributed over three different processes. So the time for this phase is the maximum of all the processes.

For the weightage calculation part, we used matrix multiplications to parallelize computations. The basic idea is that while finding the dot product of two matrices, only one row and one column affect the corresponding output value. We formulated the problem in matrix multiplication and performed dot product over the matrices. The dimensions of different matrices and the steps are shown below.

$$
y_{\text{base}} = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1k} \\ y_{21} & y_{22} & \cdots & y_{2k} \\ \vdots & \cdots & \ddots & \vdots \\ y_{n1} & y_{n2} & \cdots & y_{nk} \end{bmatrix}
$$

---

[1] https://scikit-learn.org/stable/.

$$\text{weights} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_k \end{bmatrix}$$

$$y_{c1} = y_{\text{base}_{c1}} \cdot \text{weights}$$

$$y_{c2} = y_{\text{base}_{c2}} \cdot \text{weights}$$

$$\vdots$$

$$y_{cm} = y_{\text{base}_{cm}} \cdot \text{weights}$$

$$y_{\text{concat}} = [y_{c1} \quad y_{c2} \quad \cdots \quad y_{cm}]$$

$$y_{\text{final}} = \max(y_{\text{concat}})$$

where,

n is the number of samples,

k is the number of models,

$c1 \ldots cm$ are the different classes,

y_{base} is a matrix of size $(n \times k)$ and obtained by individual models' predictions,

weights is a matrix whose elements are normalized as discussed above,

$y_{\text{base}_{ci}}$ is a matrix derived from y_{base} whose values are predicted as class ci,

y_{concat} is the matrix obtained by stacking $y_{\text{base}_{ci}}$ side-by-side,

y_{final} is the maximum of every row of y_{concat}.

4 Experimentation and Results

4.1 Data Description

To verify our results, we used two benchmark datasets [10, 11]. HUST dataset [10] was used by the base paper. We followed the same approach for data-preprocessing, like loading the images as gray scale, resizing the images into (150, 150) pixel and using min-max normalization to bring the scale of different images between 0 and 1. We split the dataset in the ratio of 9:1. Some distribution of data is shown in Table 1.

Table 1 Data statistics of different datasets

Dataset	nCT	niCT	pCT
HUST [10]	9979	5705	4001
Mendeley [11]	2628	–	5427

4.2 Metrics Used

We used a range of different metrics other than accuracy like: $f1$-score, precision, recall (sensitivity) and specificity. The formulas for calculating each of the mentioned metrics are given below.

$$\text{speedup} = \frac{\text{time}_{\text{sequential}}}{\text{time}_{\text{parallel}}}$$

$$\text{efficiency} = \frac{\text{speedup}}{n_{\text{processors}}}$$

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

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{recall (sensitivity)} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

$$f1\text{-score} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

where,
TP stands for true-positive,
FP stands for false-positive,
TN stands for true-negative,
FN stands for false-negative.

4.3 Performance Analysis

We performed all the experiments to note consistent results, including duplicating the base paper on an i5 octacore 8 GB RAM machine. The time taken by different approaches (sequential and parallel) with different weighted metrics on the two datasets is given in Table 2. The improvement in the time taken between sequential and parallel approaches is by matrix multiplication. The time taken to predict the base predictions is the time taken by the longest processor. The time taken to execute the parallelizable part in the sequential model is approximately **12 s**. In contrast, the proposed method was able to execute it in just **0.00035 s** on average with Mendeley data. Similarly, the sequential approach took around **73 s**. In contrast, the parallel approach took only **0.00098 s** on average with HUST data. The k-NN algorithm took most of the time as it compares the input image to all the training data and selects the data with a minimum distance metric (879 s).

Speedup and Efficiency on HUST Data The k-NN model took 879 s to calculate the base predictions. So it is split into three processes which now take 293 s each. So the time taken for the base predictions phase is 293 s. The time taken to compute weighted prediction is 0.00098 s. So the speedup is calculated as:

$$\text{speedup} = \frac{1233.314}{293.556} = \textbf{4.201}$$

$$\text{efficiency} = \frac{4.201}{7} = \textbf{0.6}$$

Speedup and Efficiency on Mendeley Data The k-NN model took around 561 s to calculate the base predictions. So it is split into three processes which now take 187 s each. So the time taken for the base predictions phase is 187 s. The time taken to compute weighted predictions is 0.00035 s. So the speedup is calculated as:

$$\text{speedup} = \frac{764.427}{187.415} = \textbf{4.079}$$

$$\text{efficiency} = \frac{4.079}{7} = \textbf{0.583}$$

Since k-NN is distributed among three processes and the remaining base models are run in single processor each, there are 7 processors in total.

4.4 Comparison of Results

Figure 2 shows the confusion matrices for different metric-normalization approaches on HUST dataset [10]. We see that there are no changes in the confusion matrices

Table 2 Execution time by the proposed model on the two datasets

Dataset	Mode of execution	Weighted metric	Time taken (in s)
HUST	Sequential	Accuracy	1233.314
	Parallel	Accuracy	293.556
		Precision	293.556
		Recall	293.556
		Specificity	293.561
		F1-score	294.557
Mendeley	Sequential	Accuracy	764.427
	Parallel	Accuracy	187.415
		Precision	187.415
		Recall	187.415
		Specificity	187.416
		F1-score	188.015

(a) Accuracy (base paper)   (b) Accuracy   (c) Precision

(d) Sensitivity   (e) Specificity   (f) F1-score
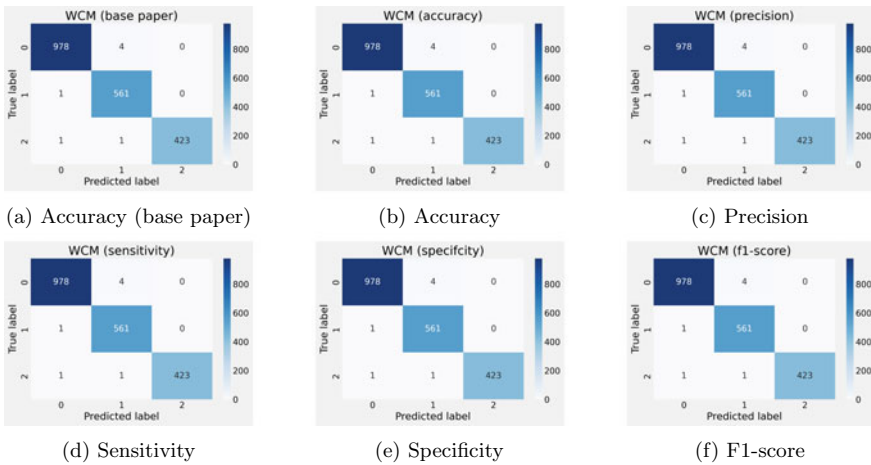
Fig. 2 Confusion matrices of different metric-normalization approaches on HUST dataset

as all the models are on par with similar values for all metrics. So the weightage is almost equally distributed with small absolute changes. Since we used only five models to dominate a class, the weightage should be high enough. So we introduced a dummy model that always classifies an image based on the frequency. The confusion matrices for the HUST and Mendeley datasets are shown in Figs. 2 and 3 respectively.

(a) Accuracy (base paper)   (b) Accuracy   (c) Precision

(d) Sensitivity   (e) Specificity   (f) F1-score
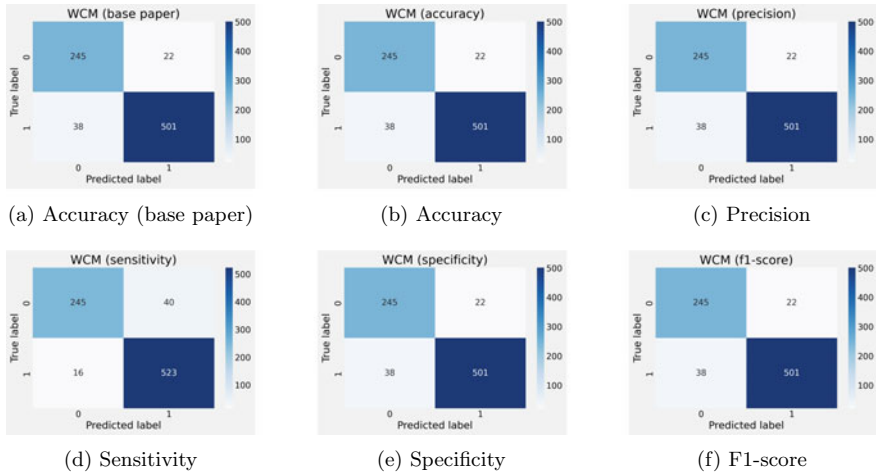
Fig. 3 Confusion matrices of different metric-normalization approaches on Mendeley dataset

5 Conclusions and Future Scope

This paper produced an improved approach to reduce the testing time and metric-normalization comparison to find the best metric to normalize and find the weights. The proposed method works exceptionally well and fast. However, due to the quality of the data, all models are trained with high and similar accuracies. If some models perform poorly on the data, this will show robust and reliable results.

In the future, we want to extend the work with other models with less predicting time and try to parallelize even more to compute predictions for millions of images in under 1 s and ease the load on the doctors and prepare for future outcomes.

References

1. Bondugula RK, Udgata SK, Bommi NS (2021) A novel weighted consensus machine learning model for covid-19 infection classification using CT scan images. Arab J Sci Eng 1–12
2. Wang L, Lin ZQ, Wong A (2020) Covid-net: a tailored deep convolutional neural network design for detection of covid-19 cases from chest X-ray images. Sci Rep 10(1):1–12
3. Wang S, Kang B, Ma J, Zeng X, Xiao M, Guo J, Cai M, Yang J, Li Y, Meng X et al (2021) A deep learning algorithm using CT images to screen for corona virus disease (covid-19). Eur Radiol 1–9
4. Narin A, Kaya C, Pamuk Z (2021) Automatic detection of coronavirus disease (covid-19) using X-ray images and deep convolutional neural networks. Pattern Anal Appl 1–14
5. Zhang Y-D, Satapathy SC, Zhu L-Y, Górriz JM, Wang S-H (2020) A seven-layer convolutional neural network for chest CT based covid-19 diagnosis using stochastic pooling. IEEE Sens J
6. 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 Fract 142:110495

7. Zhang Y-D, Satapathy SC, Zhang X, Wang S-H (2021) Covid-19 diagnosis via densenet and optimization of transfer learning setting. Cogn Comput 1–17
8. Sun Y, Wong AKC, Kamel MS (2009) Classification of imbalanced data: a review. Int J Pattern Recogn Artif Intell 23(04):687–719
9. Ferri C, Hernández-Orallo J, Modroiu R (2009) An experimental comparison of performance measures for classification. Pattern Recogn Lett 30(1):27–38
10. Ning W, Lei S, Yang J, Cao Y, Jiang P, Yang Q, Zhang J, Wang X, Chen F, Geng Z et al (2020) iCTCF: an integrative resource of chest computed tomography images and clinical features of patients with covid-19 pneumonia
11. El-Shafai W, Abd El-Samie F (2020) Extensive covid-19 X-ray and CT chest images dataset. Mendeley data, vol 3

An Integrated Approach to Reduce Class Imbalance

B. Manjula and Shaheen Layaq

Abstract The most challenging task in the software development life cycle is to defect management as early as possible to increase reliability. Most of the available defect predicting methods suffer from class imbalance problem. In this paper, classification is done after resolving the class imbalance by balancing defects and non-defect records. Our proposed algorithm class imbalance reduction using K-nearest neighbor (CIRKNN) algorithm is contrast to synthetic minority oversampling technique (SMOTE) which is considered as base and most popular machine learning algorithm for handling class imbalance problem. The basic concept behind SMOTE is that it tries to balance the majority and minority classes by picking the random sample from minority class. But, drawback is that randomly selecting initial sample may not be appropriate. So, our proposed algorithm CIRKNN integrates both centroid and K-nearest neighbor (KNN) concepts to balance the datasets. To evaluate performance from the repository of PRedict or Models in Software Engineering (PROMISE), open-source software defective datasets of about forty were collected and passed to CIRKNN for balancing; after balancing, these were passed to eight different classifiers, and six measures of performance were also done. By cross-examining the forty-eight accuracy values generated by CIRKNN and SMOTE, we observed that thirty-eight values of CIRKNN are showing good and improved performance, and ten are showing equal performance. Finally, our algorithm outstands in overall performance and showed good accuracy than SMOTE.

Keywords Classification · Class imbalance · Oversampling · Software defect prediction

1 Introduction

Today, the whole universe is moving toward digitalization. To perform digital transaction, humans are focusing on software-based system. The accuracy of software-based

B. Manjula · S. Layaq (✉)
Department of Computer Science, Kakatiya University, Warangal, Telangana, India
e-mail: mdshaheen1905@gmail.com

application depends upon its quality [1]. The quality of software application is good if it contains no defects. If it contains defects, maintaining that defects becomes costly, waste of time, and it cannot withstand in the market. It is better to use software defect prediction methods [2] to identify the defects at an early stage and to overcome the defects in less cost. Machine learning algorithms contain many defects prediction methods, but most of them suffer with the class imbalance problem. The results generated by machine learning algorithm in the past do not show good accuracy if it contains problem of imbalance in a particular class [3]. When a particular class contains greater observations than that of other class, then class imbalance occurs. Here, the greater observation class is represented as the majority class, and the lesser observation class is represented as the minority class. When such imbalance dataset is passed to classification, we get an inaccurate result, and it may lead to misclassification. So, there is a need to balance between two classes, and later classification has to be done. Figure 1 shows a software application dataset. The software application dataset consists of non-defect (ND) and defect (D) data samples, and if we observe, the non-defect samples are more than that of the defect samples. The more count of the non-defect samples forms the majority class, and less count of the defect samples forms the minority class.

Now, here our software application dataset is facing an imbalance problem. At this point if we perform classification on this dataset, we may lead to wrong classification as the outcome will be in support of the greater observation class (the non-defect class). In Fig. 1, it contains less count of the defect samples, but in terms of cost, there are more valuable. To get accurate result, the imbalance defect and non-defect data must be balanced, and later classification has to be done to get good accuracy result.
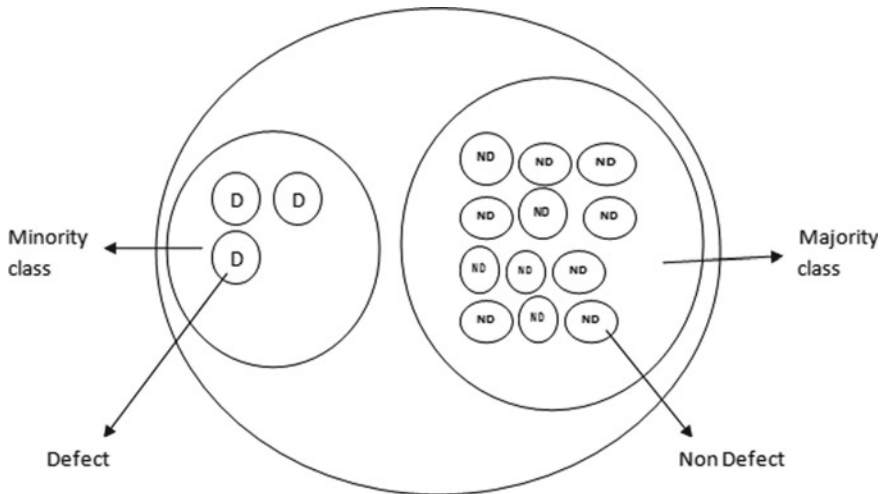

Fig. 1 Software application dataset

The diverse sampling methods are suggested and used to convert imbalance dataset to balance dataset. By making dataset as balance, its accuracy will increase.

1.1   Contribution of Research

1. A complete new framework was designed. Firstly, we collected imbalanced data related to software defect from PROMISE repository.
2. We proposed a new integrated method known as class imbalance reduction using K-nearest neighbor (CIRKNN). In this proposed algorithm, we integrated the concept of central tendency and K-nearest neighbor. New samples were created between them till defect and non-defect class are balanced.
3. After making data as balance using integrated method CIRKNN, it is passed to perform classification, and performance was calculated.
4. The calculated performance of CIRKNN was compared with the SMOTE, and CIRKNN showed better performance than SMOTE.

2   Associated Work

The class imbalance problem is addressed with many machine learning algorithms by taking software defect prediction (SDP) as basic application. The class imbalance in SDP is considered by various methods which are categorized as data based and algorithm based.

The *data-based method* modifies observations either by incrementing or decrementing. The methods such as oversampling randomly, undersampling randomly, and SMOTE come under it. The cons of data-based method are losing or replicating of some important data.

The best *algorithm-based methods* [4] are ensemble and cost sensitive. Bagging and boosting come under *ensemble methods* which try to resolve the class imbalance present in SDP. In *cost-sensitive* [5] (CS) method, different misclassification costs are assigned to different classes. In SDP, the defective observations are assigned with big misclassification cost, and non-defective observations are assigned with small misclassification cost. By which the defective observations are been identified. The cost boosting method for SDP was proposed by Khoshgotaar et al. [6]. Cost-sensitive boosting using neural networks for SDP was proposed by Zheng [7]. Arar and Ayan [8] studied CS neural network. The two classifiers such as CS feature selection and CS neural network were present in two stages CS learning for SDP proposed by Liu et al. [9]. The dictionary learning and random forest [2] were combined with CS by some researchers. CS was also used in the CPDP scenario [10]. Whatever it might be setting of cost value for CS is still a big problem. The weighted least squares twin support vector machine (WLSTSVM) was used to developed SDP system by Omar and Agarwal [11], where greater value was assigned to defective observation and

smaller value to class with non-defective observation. The famous KMFOS method introduced by Gong et al. [12] creates different observations in the defective space. In KMFOS, the 'K' number of clusters was created for defective samples using K-means clustering method, and later new observations were created using interpolation. Eight different classifiers were used by Sohan et al. [13] to overcome class imbalance problem in CPDP. Song et al. [14] conducted various experiments and concluded that for highly imbalanced software defect prediction datasets, imbalanced learning could be considered. To evaluate the inconsistency between imbalanced dataset and balanced dataset, Sohan et al. [15] stated that SDP gets affected if there is no balance between the class with defective and non-defective observations. Whatever might be CS methods still faced the problem of assigning correct cost and ensemble methods faced problem with increase of classifiers as complexity increases.

3   CIRKNN: An Integrated Method

Our integrated method CIRKNN can be explained with a proposed framework as shown in Fig. 2. Here the imbalance dataset is collected first from the real world, and from which minority data samples are selected and passed to SMOTE and CIRKNN
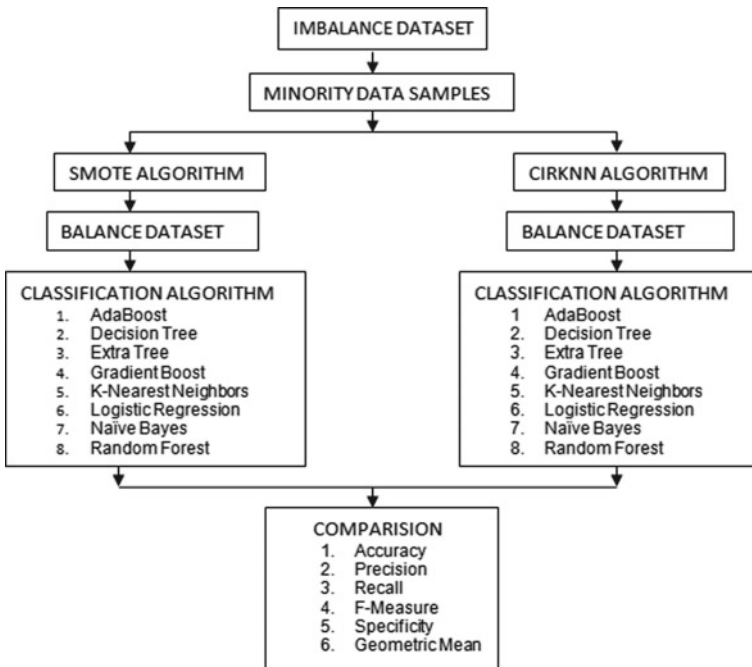
Fig. 2 Proposed framework

algorithm. In turn, both algorithms generate balanced dataset which are passed to eight classification methods, and later performance of both is calculated.

3.1 Imbalance Dataset and Minority Data Samples

Firstly, the imbalanced defective dataset has to be taken. Now, let us denote that imbalanced defective dataset as IMDS where it consists of set of 'm' records which is shown in Eq. 1.

$$IMDS = \{b1, b2, b3, \ldots, bm\} \tag{1}$$

where each b_i lies between 1 and m values ($1 \leq i \leq m$). Here, each record represents the particular component in the application, and it consists of attributes (software measure) of size 'n'. To keep count of bugs, an extra attribute bug is used as class label. If the counter value of bug is greater than zero, it is the defective module, and if bug value is equal to zero, it is non-defective module. To represent this 0 or 1 value, a little bit of preprocessing is required. Divide this imbalance defective dataset into two classes: defective (minority) and non-defective (majority) class. Let us denote defective class dataset as D_m and non-defective class dataset as D_M, respectively. Now, minority class dataset (Dm) is passed to both SMOTE and CIRKNN algorithm.

3.2 SMOTE Algorithm

SMOTE has shown best performance in balancing the imbalance dataset, and it is widely used. It selects a random sample from minority class and finds its nearest neighbor. Then new random observations or synthesized data are created in between the chosen random observation and its nearest neighbor. An algorithm for SMOTE process can be explained in Algorithm 1.

Algorithm 1 SMOTE_PROCESS

1. Collect the imbalanced dataset and identify the minority class.
2. A random sample is selected from the minority class, its nearest neighbor is known, and difference between them is calculated.
3. Generate random number between 0 and 1 and multiply with the difference which is calculated in step 2.
4. By adding the random number to the feature vector, identify the new point.
5. Repeat steps 3 to 4 for all feature vectors.

Drawbacks of SMOTE are as follows:

1. If random number is not selected properly, the whole accuracy reduces.
2. Space and time complexity is high.

3. It is sensitive to outliers.
4. As training data size grows, system becomes slower.
5. It does not perform effectively as dimensionality increases.

3.3 CIRKNN Algorithm

The CIRKNN is our proposed integrated algorithm. It takes minority class samples and calculates the centroid. From that center sample, it calculates the K-nearest neighbor. The working model of CIRKNN is shown in Fig. 3.

After calculating K-nearest neighbor, it arranges all the KNN in increasing order and picks first sample with least distance from the centroid. Later, new samples have been generated in between the centroid and first nearest neighbor by using following Eq. 2.

$$\text{(distance from centriod to first Nearest Neighbour)} \over 3 + \text{(random number between 0 and 1)} * \text{centroid} \quad (2)$$

After generating new samples, these new samples are adjoined with minority class. Same task is redone for the second nearest neighbor to the centroid, and this process repeats until the majority and minority class samples are balanced. Here, the biggest task is selecting of the 'k' value for KNN. It has to be selected properly neither too small nor too big. There is no basic proper method for selecting an exact value for 'k'. So, a 'k' value has to be selected randomly and dependent on the amount of oversampled (new) data required. In our work, we have taken the 'k' value as five, and it showed good accuracy. An algorithm for CIRKNN process can be explained in Algorithm 2.

Fig. 3 Demonstration of proposed CIRKNN method
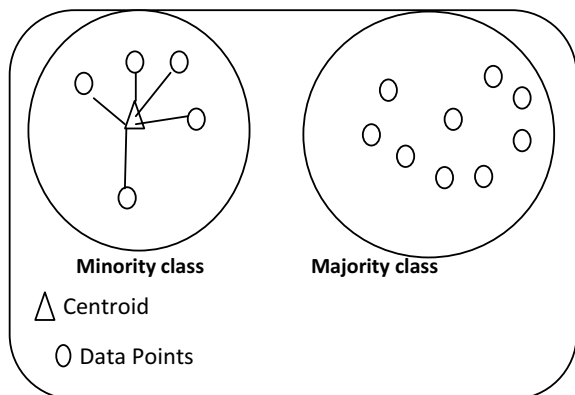
3.4 Balance Dataset

From the generated balance dataset, we have considered 20% data records for testing, and the 80% of left records were assumed as training data and passed to the classifiers.

3.5 Classification Algorithms

For testing our work performance, the best eight classifiers have been selected which are mentioned below.

AdaBoost. AdaBoost (AB) [16] is popularly known as ensemble machine learning method. To improve performance of low classifiers, it combines all low performing classifiers into one.

Decision tree. One of the supervised learning techniques is decision tree [17] (DT). It is a tree structured classifier. It consists of internal nodes, branches, and leaf nodes. Initially, the root node is considered for classification. The root value is compared with other values, and branching or jumping is performed. At next round, the values are compared with sub-node and move further. The process ends when data values are completed, or it reaches the leaf node of the tree.

Extra Tree. Extremely randomized trees [18] (ET) classifiers are ensemble learning technique. The multiple-related decision trees present in the forest are aggregated. From the original training sample, each decision tree is constructed. The construction processes of the decision tree and random forest are different.

Gradient Boosting. Gradient boosting (GB) [19] uses a weak prediction model to generate a classification model. It uses stage-wise method to create the final model. To perform the operation, it requires three things: a loss function to be optimized, for prediction a weak learner, and to add weak learners to minimize the loss function, an additive model is required.

Algorithm2: Class Imbalance Reduction using KNN(CIRKNN)

Input: Imbalanced Dataset (IMDS) with a_1, a_2, a_3, .., a_m attributes which represent label of class and b_1, b_2, b_3,..., b_m are records

Output: Balanced Dataset(BDS) with balance number of non-defect and defect records

Step-1: Depending on class label value IMDS is divided into two classes defect and non defect respectively.

Step-2: Class with less count of records is represented as minority class(D_m)

Step-3: Class with more count of records is represented as majority class(D_M)

Step-4: Using minority class(D_m) centroid (center) is calculated

      center={mean(a_1),mean(a_2),mean(a_3),...,mean(a_m)}

Step-5: For each record b_i in D_m

      Step-5.1: using Euclidian distance calculate the distance between b_i and center

          dist1(b_i, center).

Step-6: According to distance sort the records in ascending order.

Step-7: Choose the first k records with least distance(Dist1$_{least}$)

Step-8: for j ← 1 to k

    Random numbers r_1, r_2, $r_{3,4,...}$, r_n are generated which lies in between 0 and 1,

    Where n=| D_M |-| D_m | by which non defect and defect records are balance.

      Step-8.1: For each random number r_s

          Step-8.1.1: A new record is generated as Dist1$_{least}$ + r_s * center

          Step-8.1.2: Generated new record is added to D_m

K-Nearest Neighbor (KNN). KNN [20] is a simplest supervised machine learning technique. Its strong assumption is that all the related observations are very close to each other. It selects the *K*-neighbor depending on their distance where K is the user specified parameter.

Logistic Regression. One of the mostly used statistical learning techniques is logistic regression [21] (LR). It is used when more than one independent variable values are present. It tries to find the connection between the set of independent variables and class label.

Naive Bayes Classifiers. Naive Bayes [22] (NB) classification is a machine learning model which is dependent on the basic model of Bayes formula. Here, it is supposed that the features are not dependent. It finds the chance of occurring an event *A* when an event *B* has occurred. Naive Bayes can be given by Eq. (3)

$$p(H|M) = \frac{P\{M|H\}P(H)}{P(M)} \tag{3}$$

where

$P(H)$ is the prior probability of class *H*.

$P(M)$ is the prior probability of data instance *M*.

$p\left(H/M\right)$ is the possibility of *H* when *M* stated.

$P\left(M/H\right)$ is the posterior probability of data instance *M* on class *H*.

Table1 Confusion matrix

	Predicted negative	Predicted positive
Actual negative	TN	FP
Actual positive	FN	TP

Random Forest. Random forest [23] (RF) is a well-known machine learning technique. All the decision trees have been combined and form a random forest. While splitting a node, it takes only a random subset feature into account. Replacement sampling is used in the random forest; while testing all the trees, prediction is considered; later their average is performed; the final prediction has been done.

3.6 Comparison

To evaluate the performance, six classifier measures have been used which are shown by Eq. 4–9, and matrix of confusion is used shown by Table 1.

$$\text{Sensitivity or Recall} = \frac{\text{True positive}}{\text{True positive} + \text{False Negative}} \tag{4}$$

$$\text{Specificity} = \frac{\text{True Negative}}{\text{True Negative} + \text{False Positive}} \tag{5}$$

$$\text{Geometric Mean} = \sqrt{\text{Sensitivity * Specificity}} \tag{6}$$

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \tag{7}$$

$$F\text{ - Measure} = \frac{2* \text{ Recall* Precision}}{\text{Recall} + \text{Precision}} \tag{8}$$

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negaive}}{\text{True Positve} + \text{True Negative} + \text{False Positive} + \text{False Negative}} \tag{9}$$

4 Experimental Results

From the tera-PROMISE repository [24], forty open-source datasets were considered which were related to software defect prediction, and in which, minority class percentage was found very less. Each dataset consists of twenty software metrics. The comparative result analyzing CIRKNN over SMOTE by using six performance

Table 2 Comparative result of CIRKNN over SMOTE

Classifier	Accuracy		Precision		Recall	
	SMOTE	CTRKNN	SMOTE	CIRKNN	SMOTE	CTRKNN
AB	0.83 ± 0.11	0.85 ± 0.10	0.85 ± 0.12	0.87 ± 0.12	0.82 ± 0.12	0.82 ± 0.11
DT	0.82 ± 0.10	0.83 ± O.ll	0.82 ± 0.10	0.82 ± 0.13	0.83 ± 0.12	0.83 ± 0.11
et	0.79 ± 0.12	0.82 ± 0.11	O.77 ± 0.13	0.82 ± 0.12	0.80 ± 0.13	0.84 ± 0.11
CB	0.83 ± 0.12	0.84 ± 0.10	0.83 ± 0.12	0.85 ± 0.11	0.81 ± 0.15	0.83 ± 0.11
KNN	0.81 ± 0.12	0.82 ± 0.12	0.82 ± 0.14	0.85 ± 0.12	0.79 ± 0.15	0.79 ± O.14
LK	0.80 ± 0.11	0.81 ± 0.10	0.80 ± 0.10	0.81 ± 0.11	0.78 ± 0.15	0.80 ± 0.14
NB	0.72 ± 0.12	0.78 ± 0.11	0.80 ± 0.14	0.87 ± 0.11	0.60 ± 0.25	0.65 ± 0.22
RF	0.85 ± 0.10	0.87 ± 0.09	0.89 ± 0.11	0.89 ± 0.10	0.87 ± 0.10	0.86 ± 0.11
Classifier	F-measure		Specificity		Geometric mean	
	SMOTE	CTRKNN	SMOTE	CIRKNN	SMOTE	CIRKNN
.AB	0.83 ± 0.11	0.84 ± 0.10	0.86 ± 0.12	0.87 ± 0.11	0.84 ± 0.11	0.85 ± 0.10
DT	0.82 ± 0.10	0.82 ± 0.12	0.81 ± 0.11	0.82 ± 0.12	0.82 ± 0.10	0.83 ± 0.11
ET	0.78 ± 0.12	0.83 ± 0.11	0.78 ± 0.12	0.81 ± 0.13	0.79 ± 0.12	0.82 ± 0.11
CB	0.82 ± 0.13	0.84 ± 0.10	0.85 ± 0.11	0.86 ± 0.11	0.83 ± 0.12	0.84 ± 0.10
KNN	0.80 ± 0.13	0.82 ± 0.12	0.83 ± 0.14	0.86 ± 0.12	0.81 ± 0.12	0.82 ± 0.12
LR	0.80 ± 0.12	0.80 ± 0.11	0.82 ± 0.10	0.82 ± 0.09	0.80 ± 0.11	0.81 ± 0.10
NB	0.66 ± 0.19	0.73 ± 0.18	0.84 ± 0.15	0.91 ± 0.10	0.68 ± 0.16	0.75 ± 0.14
RF	0.85 ± 0.09	0.85 ± 0.09	0.89 ± 0.10	0.90 ± 0.09	0.87 ± 0.08	0.87 ± 0.09

measures with eight classifiers is shown as 'mean ± standard deviation', and it is given in Table 2.

By the cross-examining the values of Table 2, we observed that out of the forty-eight values, thirty-eight values of CIRKNN are showing good and improved performance, and ten are showing equal performance. Finally, we can conclude that CIRKNN shows better accuracy than SMOTE.

5 Conclusion

Our paper has implemented CIRKNN algorithm which tried to reduce imbalance present in dataset. We have taken SMOTE as our base algorithm, and we had compared our results of CIRKNN with SMOTE. Basically, SMOTE is a machine learning and most popular algorithm used to balance the imbalance data. SMOTE tries to randomly pick a sample data from the minority class, tries to identify the nearest neighbor, and generates randomly new data from those two points. But SMOTE accuracy may not be accurate or constant because SMOTE accuracy depends on effectively choosing of the initial random sample. Our proposed algorithm overcomes it by

calculating centroid from the given minority class, and from that center data sample, it calculates KNN samples and tries to generate randomly new data between the center sample and nearest neighbor till it balances both defective and non-defective classes. We did comparative result analyzing of CIRKNN over SMOTE by using six performance measures, geometric mean, precision, accuracy, recall, specificity, and F-measure, after passing to eight best machine learning classifiers, extra tree, random forest, decision tree, K-nearest neighbor, gradient boosting, logistic regression, Naive Bayes, and AdaBoost. By cross-examining the accuracy values generated by CIRKNN and SMOTE, we observed that out of the forty-eight values, thirty-eight values of CIRKNN are showing good and improved performance, and ten are showing equal performance. Finally, our algorithm outstands in overall performance and showed good accuracy than SMOTE. For future enhancement, we can combine our approach with optimal techniques by which it can become more integrated and more accurate to obtain optimal solution.

References

1. Arora I, Tetarwal V, Saha A (2015) Open Issues in software defect prediction. Proc Comput Sci 46:906–912
2. Siers MJ, Zahidul Islam M (2015) Software defect prediction using a cost sensitive decision forest and voting and a potential solution to the class imbalance problem. In: Information systems, vol 51, pp 62–71
3. Japkowicz N, Stephen S (2002) The class imbalance problem: a systematic study. Intel Data Anal 6(5):429–449
4. Benjamin W, Nathalie J (2008) Boosting Support Vector Machines for Imbalanced Data Sets. Found Intel Syst 4994:38–47
5. Nikolaou N, Edakunni N, Kull M, Flach P, Brown G (2016) Cost sensitive boosting algorithms: do we really need them? Mach Learn 104(2–3):359–384
6. Khoshgoftaar TM, Geleyn E, Nguyen L, Bullard L (2002) Cost-sensitive boosting in software quality modelling. In: Proceedings of the 7th IEEE international symposium on high assurance systems engineering, Tokyo, Japan, pp 51–60
7. Zheng J (2010) Cost-sensitive boosting neural networks for software defect prediction. Exp Syst Appl 37:4537–4543
8. Arar ÖF, Ayan K (2015) Software defect prediction using cost-sensitive neural network. Appl Soft Comput 33:263–277
9. Liu M, Miao L, Zhang D (2014) Two-stage cost-sensitive learning for software defect prediction. IEEE Trans Reliab 63:676–686
10. Ryu D, Jang JI, Baik J (2017) A transfer cost-sensitive boosting approach for cross-project defect prediction. Softw Qual J 25:235–272
11. Omar, TD, Agarwal S (2016) Prediction of defective software modules using class imbalance learning. Appl Comput Intell Soft Comput 1–12
12. Gong L, Jiang S, Jiang L (2019) Tackling class imbalance problem in software defect prediction through cluster-based over-sampling with filtering. IEEE Access 7:145725–145737
13. Sohan, MF, Jabiullah MI, Rahman SSMM, Mahmud SH (2019) Assessing the effect of imbalanced learning on cross-project software defect prediction, pp 1–6
14. Song Q, Guo Y, Shepperd M (2019) A Comprehensive investigation of the role of imbalanced learning for software defect prediction. IEEE Trans Softw Eng 45:1253–1269

15. Sohan MF, Kabir MA, Jabiullah MI, Rahman SSMM (2019) Revisiting the class imbalance issue in software defect prediction. In: Proceedings of the 2019 international conference on electrical, computer and communication engineering, pp 1–6
16. Freund Y, Schapire R (1995) A decision-theoretic generalization of on-line learning and an application to boosting. J Comp Syst Sci 55:119–139
17. Quinlan JR (1993) C4.5: programs for machine learning. Morgan Kaufmann Publishers Inc., San Francisco
18. Geurts P, Ernst D, Wehenkel L (2006) Extremely randomized trees. Mach Learn 63:3–42
19. Friedman J (2001) Greedy function approximation: a gradient boosting machine. Ann Stat 29:1–39
20. Altman NS (1992) An introduction to kernel and nearest-neighbor nonparametric regression. Am Stat 46:175–185
21. Peng CYJ, Lee KL, Ingersoll GM (2002) An introduction to logistic regression analysis and reporting. J Educ Res 96:3–14
22. Rish I (2001) An empirical study of the Naive Bayes classifier. In: Proceedings of the JCAI 2001 Workshop on Empirical Methods in Artificial Intelligence, Seattle, WA, USA, vol 3, pp 41–46
23. Liaw A, Wiener M (2002) Classification and regression by random. Forest 2:18–22
24. Ferenc R, Toth Z, Ladányi G, Siket I, Gyimóthy T (2018) A public unified bug dataset for Java. In: Proceedings of the 14th international conference on predictive models and data analytics in software engineering, Oulu, Finland, pp 12–21

Power Quality Improvement Using PV Fed Shunt Active Filter with Different MPPT Algorithm

Pidiyar Dharani Deepika, Pratap Sekhar Puhan, and Ravikanth Mallajoshula

Abstract PV fed shunt active filter plays an important role in harmonics elimination in distribution systems. In this paper, a PV fed shunt active filter is designed for power quality improvement in a distribution system. At first, a nonlinear load is connected to the system and harmonics are generated. PV fed shunt active filter is using perturb and observes (P&O) maximum power point tracking (MPPT) and particle swarm optimization (PSO)-based MPPT techniques that extract the maximum power, and accordingly, shunt active filter works with PQ-based reference signal and hysteresis-based switching action techniques. Finally, it is concluded that the performance of the compensation is improved when PSO-based MPPT tracks the maximum power. The effectiveness of the techniques is verified through simulations.

Keywords Harmonics · PQ · SAPF · MPPT · PSO · Hysteresis controller

1 Introduction

Harmonic issues in the distribution system are a great concern in the recent age, as the demand of power electronics-based loads is increased in a vast manner [1–3]. The power electronics-based loads are nonlinear in nature, and when they are connected to the system, distorted current signal which constituents fundamental as well as harmonics components is drawn from the supply, and this current develops a distorted voltage waveform at the point of common coupling (PCC) where various loads are connected, and for efficient operation of these loads, pure sinusoidal voltage is required but unfortunately distorted voltage will become the inputs to the equipments;

P. D. Deepika (✉) · P. S. Puhan · R. Mallajoshula
Department of Electrical and Electronics Engineering, Sreenidhi Institute of Science and
Technology, Hyderabad 501301, India
e-mail: deepikapidiyar96@gmail.com

P. S. Puhan
e-mail: pratapsekhar@sreenidhi.edu.in

R. Mallajoshula
e-mail: ravikanthm@sreenidhi.edu.in

as a result, the performance of the equipment will be hampered rapidly. It is very crucial to maintain the voltage pure sinusoidal at PCC. Once the voltage is maintained at PCC, the performance of all the equipments will be improved. As the distorted voltage is the results of distorted current drawn by the load, so precautionary measure should be taken to compensate or filter out the nonlinear current from the system to enhance the power quality [4, 5].

Previously, different types of methods were used to mitigate these harmonics, as the research is going on at a vast rate, many new techniques are also coming up. İnitially, passive filter is used but the demerits of the passive filter are overcome by active filters and different configuration of active filters with various control mechanisms for specific types of problem is introduced [6–8]; if in a system current related power quality issues are coming, then it is advisable to go for shunt active filter but if voltage-related power quality issues are there, then it is better to go for series active filter and so on [9, 10].

The conventional source of energy is the major source in the power system but the recent development of distributed generation with renewable energy sources stands one of the best alternative for the conventional source [11]. The combination of both conventional and nonconventional power systems gives better result in terms of efficiency, operation and control [11–15]. In this type of arrangement, power supply can be maintained by the non-conventional source, if problems arise in the conventional.

In this work, PV fed shunt active filter is developed whose main aim is to eliminate the harmonics currents. The shunt active filter efficiently works when the DC link voltage will maintain the designed constant value, in order to obtain the voltage constant across the DC link, a PV source is connected; again the PV will work efficiently, when maximum power will be achieved. Basically, the effective operation of the active power filter depends on the control techniques' implementation for the generation of reference current signals and the efficient operation of the PV. Though different works are reported in this area [4, 5], MPPT with PSO [4–6] is the main work which is focussed here, and the results are compared with the P&O MPPT. Finally, the harmonics mitigation rate when these two techniques is implemented to maintain the voltage constant across the voltage source inverter, during that period the THD analysis is presented.

The research work is explained fully with six sections. In Sect. 1, a brief introduction of power quality issues and renewable energy source-based distributed generation is presented. The working model is described in Sect. 2, Sect. 3 is described briefly the different control techniques of implementation in the model. Obtained simulation results and the analysis are analysed in Sect. 5 and finally conclusions in Sect.6
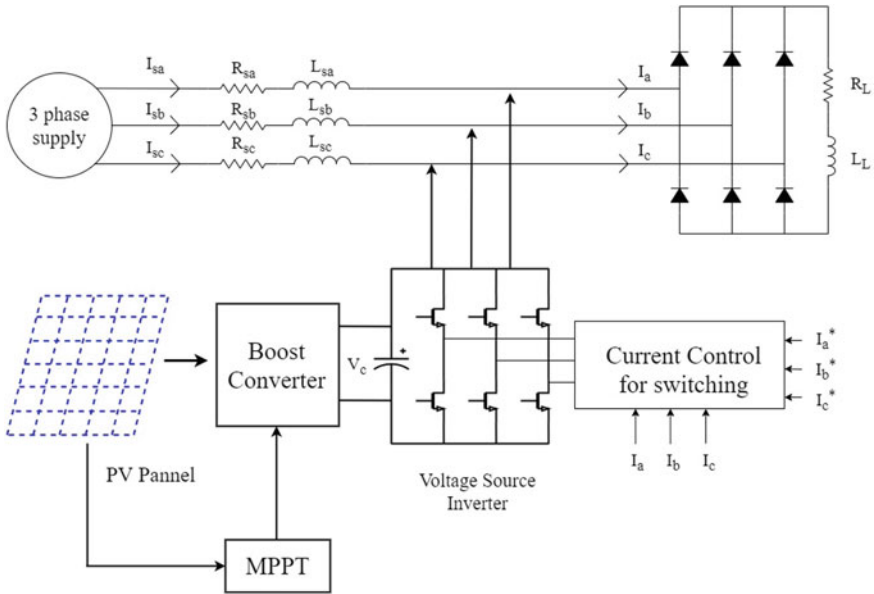
Fig. 1 Basic configuration of the system

2   Proposed PV-Based Active Filter System

The proposed working model is presented in Fig. 1. The system is mainly associated with mainly three parts, three-phase supply, a nonlinear load and a PV fed shunt active filter. The supply line has one resistor and an inductor. The nonlinear load consists of diode bridge rectifier along with R-L. The intersection of the designed nonlinear load is to produce harmonics contents by drawing nonlinear current from the system. The PV fed shunt active filter consists of IGBT, PV panel, boost converter, DC link capacitor. The main work of the PV panel is to maintain the DC voltage across the voltage source converter, because during switching operation there is a chance of losses and if the constant voltage is not maintained throughout the operation, it will affect the performance of the active filter results failure of compensation of harmonics. The maximum power point tracking is achieved through different techniques. The details of the control techniques implementation are discussed in the next section briefly.

3   Proposed Control Approach

The efficient operation of the proposed filter depends on how the control technique is implemented. At first, to keep the voltage across the capacitor constant, proper

operation of the PV panel should be achieved and it is achieved through two different MPPT algorithms, and after that, reference signal of the active filter is achieved through conventional PQ theory is discussed, and finally for switching operation, hysteresis current controller is used.

3.1 P & O MPPT Approach Towards PV

The conventional P& O MPPT techniques implementation is carried out in the work, and it is presented in Fig. 2, the corresponding flowchart is shown in Fig. 3, the total current "I" in the proposed system is taken from [6] and is presented in Eq. 1.

Fig. 2 Proposed circuit of PV system
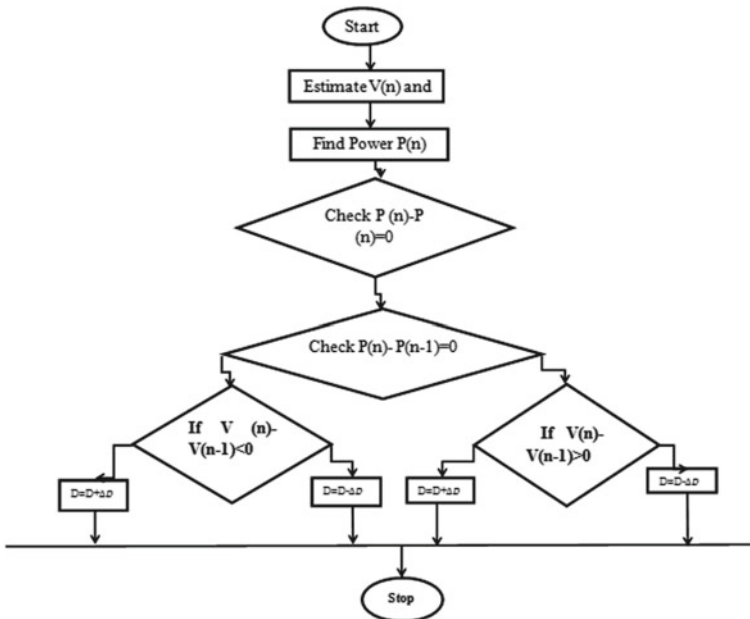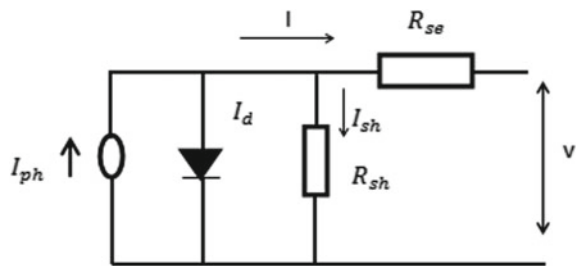
Fig. 3 Block diagram of the flowchart

$$I = I_{\text{Phi}} - I_0\big[\exp(v + \text{IR}_{\text{sh}}/nv_t)\big] - \frac{v + I_{\text{shi}}}{R_{\text{sh}}} \qquad (1)$$

At first, the popular algorithm P&O used to track maximum power point (MPP) due to its simplicity [7, 12]. The P&O algorithm is explained in the flowchart as shown in Fig. 3, [3] in this work, to track the maximum power. As per the work, IV and PV characteristics are extracted at different irradiation levels 400,600,800 and 1000 W/m^2 at 25 °C which is presented in Figs. 4 and 5, respectively. The solar power increases as radiation increases keeping temperature constant and solar power decreases as temperature vary keeping radiation constant.
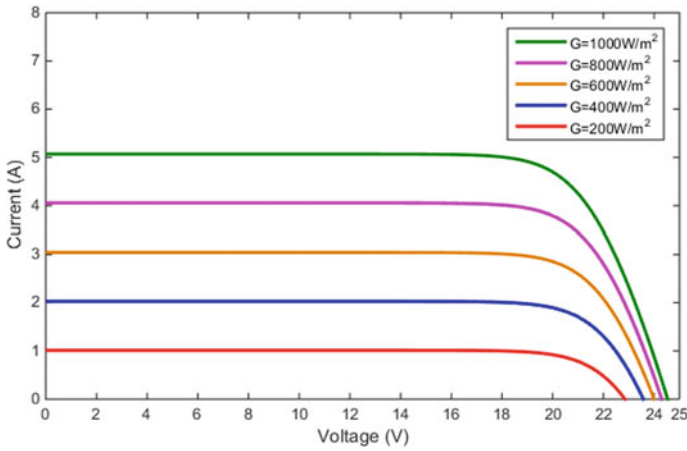
Fig.4 *V-I* characteristics of solar panel
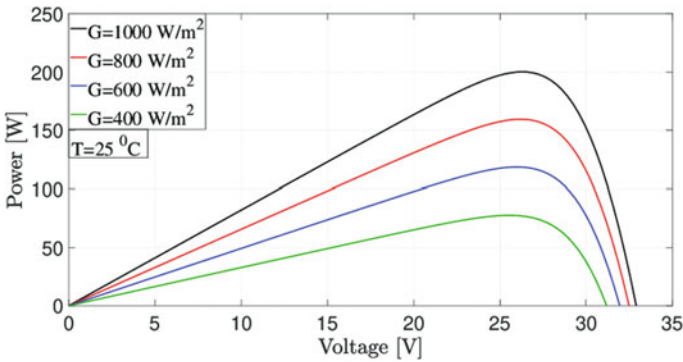
Fig. 5 *PV* characteristics of solar panel
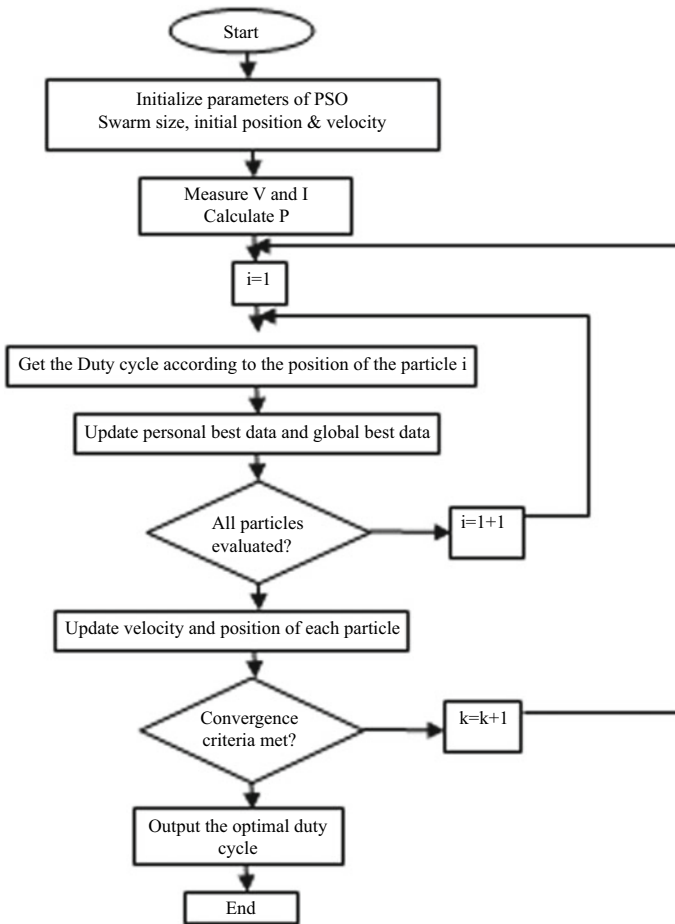
Fig. 6 Block diagram of the flowchart of PSO-based MPPT

3.2 PSO MPPT Approach Towards PV

The P&O MPPT associates with numerous problems which result in the damage of PV cell. To overcome these disadvantages, soft computing MPPT technique using PSO is proposed. By using PSO, the issues of stability and tracking efficiency in the existing P&O are overcome.

This nature-inspired evolutionary and stochastic optimization were developed in 1995 by James Kennedy and Russell C. Beernaert, and it is used in many optimization problems [4, 5]

The steps required to update the velocity and position of the particle are related to this problem and are mentioned in 2 and 3

$$V_i^{k+1} = w^k V_i^k + c_1 r_1 \left(P_{\text{best},i}^k - x_i^k \right) + c_2 r_2 \left(G_{\text{best},i}^k - x_i^k \right) \tag{2}$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \tag{3}$$

where
i—variation of the optimization vector.
k—number of iterations.
w—is the inertia weight.
V_i^k and x_i^k—velocity and position of the ith variable for kth iteration.
Coefficients $c1$ and $c2$—acceleration parameters and their values are usually close to 2.

The variable $P_{\text{best},i}^k$ saves the best position reached by the ith particle up to the exact time of measurement only when the condition declared in Eq. (4) is satisfied

$$P_{\text{best},i}^k = x_i^k \, if \, fit(x_i^k) \geq fit(P_i) \tag{4}$$

In MPPT controller, PSO is applied by taking x_i^k as duty cycle of the boost converter. The velocity is updated by using Eq. (5); for each duty cycle and PV system, output is selected as fitness function. The PSO MPPT is explained by using the flowchart as shown in Fig. 6.

$$V_i^{k+1} = w^k * V_i^k + c_1 * r_1 * (D_{\text{best},i}^k - D_i^k) + c_2 * r_2 (G_{\text{best},i}^k - D_i^k) \tag{5}$$

3.3 PQ Theory Approach Towards Reference Signal Generation of the Active Filter

Several control techniques have been used in active filter to generate the reference current signals [1–3], PQ theory is one of the techniques which is used in this work for reference current signal generation [7]. In this method, at first three-phase supply voltage is converted into two-phase using Clarke transformation; similarly, the three-phase line current also is converted into two-phase using the same transformation; after that, active power and reactive power are calculated using [10]. The active power loss component is calculated from the DC voltage error, and reactive power loss also is calculated from the source voltage using PI controller; after that, reference active and reactive power are obtained, once active and reactive references are obtained, then two-phase reference current is obtained using mathematical calculation; after that, inverse Clarke transformation is implemented to get three-phase reference current. The mathematical analysis is taken from [7], and the block diagram is presented [10] (Fig. 7).

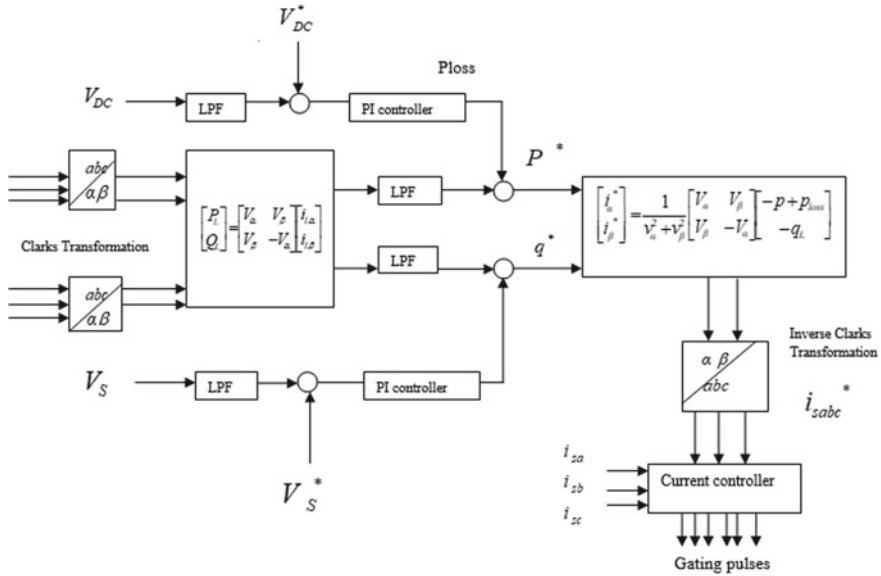The final expression for the reference source current is expressed in terms of $i_{c\alpha}^*, i_{c\beta}^*$ []

Fig. 7 Block diagram of PQ theory

$$
\begin{bmatrix} i_{sa}^* \\ i_{sb}^* \\ i_{sc}^* \end{bmatrix} = \sqrt{\frac{2}{3}} \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ \frac{1}{2} & \frac{-\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} i_{c\alpha}^* \\ i_{c\beta}^* \end{bmatrix}
\tag{6}
$$

4 Results and Analysis

The results of simulation of PV fed filter can be observed in the following figures. Proposed filter out the harmonics by using PQ theory which reduces the THD of the system and improves power quality of the system and also supplies the PV output power to the load. The PV power with P&O and PSO can be observed from Fig. 8, and we can conclude from the waveform that MPPT PSO is efficient compared to P&O, reduces the oscillations, steady-state time and operates efficiently in changing weather conditions. The convergence time is faster in the case of PSO as compared to P&O visible from the result as shown in Fig. 8.

From Fig. 9, distortion in the load current waveform due to nonlinear load can be observed. Due to the nonlinear loads, power quality of the system gets affected. To reduce the distortion and improve the waveform at $t = 0.02$, shunt active filter is activated which reduces the harmonics and improves the current waveform which can be observed in Fig. 10, and it is coming due to P&O MPPT and in Fig. 11 source current due to PSO and MPPT technique. The active and reactive power analysis of
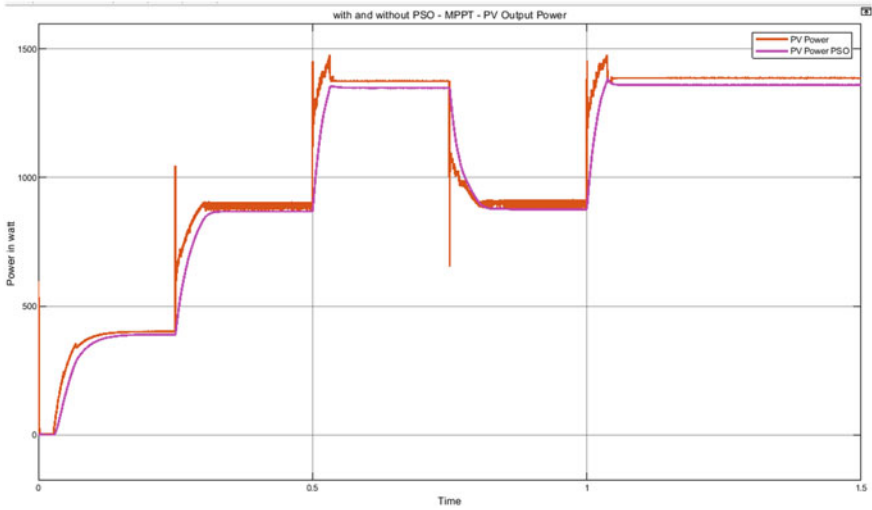
Fig. 8 PV power with P&O and PSO
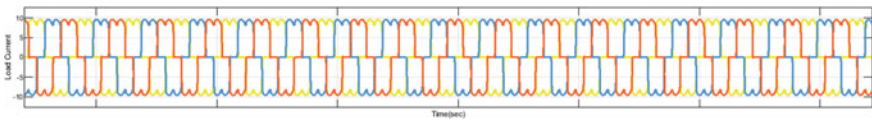
Fig. 9 Load current
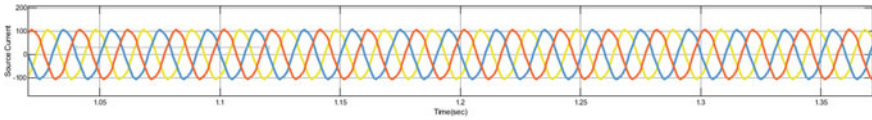
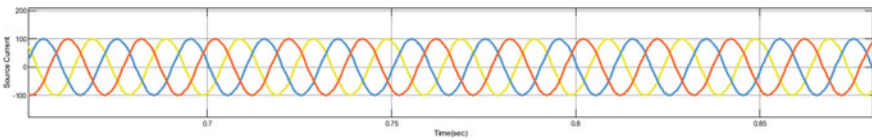**Fig. 10** Source current with P&O MPPT

Fig. 11 Source current with PSO MPPT

both the conditions is shown in Figs. 12 and 13, respectively. From Figs. 12 and 13, we can observe the active power transferred from source to load got reduced from $t = 2$ s to $t = 5$ s which indicates the active power required for the load is transferred
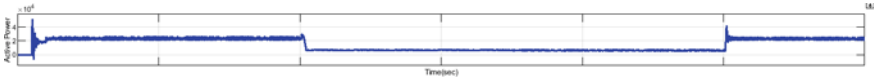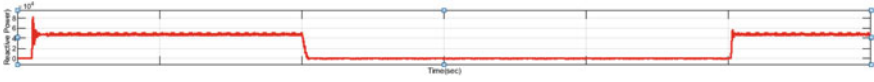
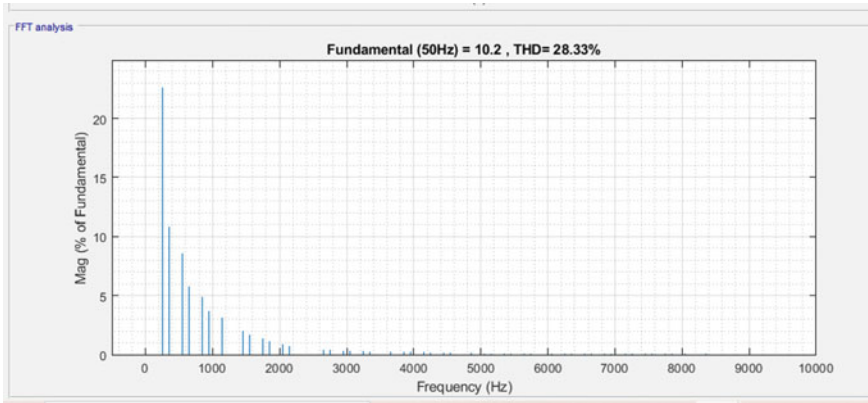**Fig. 12** Active power analysis

Fig. 13 Reactive power analysis

Fig. 14 THD analysis without filter

from the PV system, and also, the reactive power is compensated after that the filter is activated.

The analysis of power quality through total harmonics distortions (THD) is presented in Figs. 14, 15 and 16. In the presence of nonlinear load, the harmonics content is 28.33% of fundamental as shown in Fig. 14. The moment filter is activated with P& O MPPT it comes down to 4.98% of fundamental and when PSO MPPT is implemented it is 1.18% of fundamental followed in Figs. 15 and 16, respectively (Table 1).
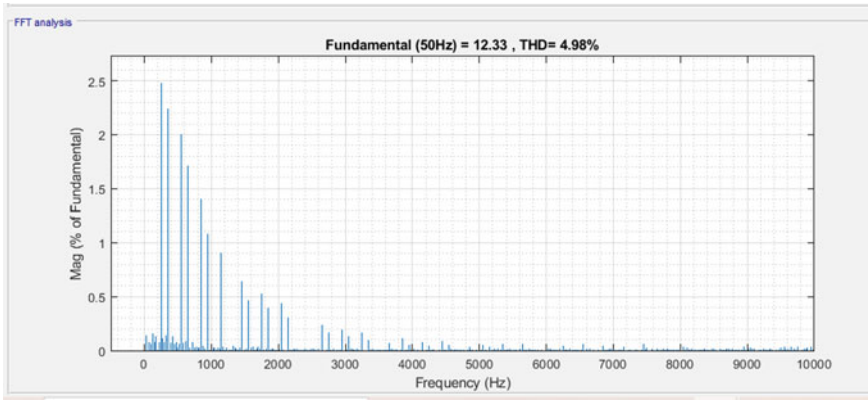
Fig. 15 THD analysis filter
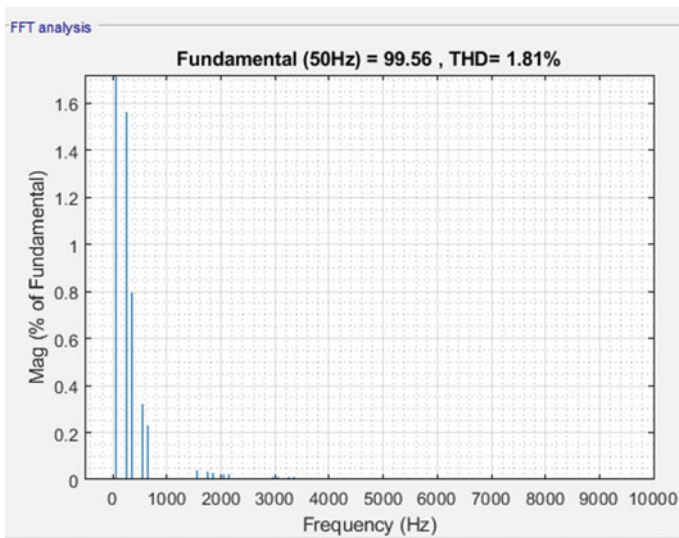
Fig. 16 THD analysis filter and PV

Table 1 System behaviour is analysed with and without filter and PV

System configuration	THD (%)
System with nonlinear load	28.33
System with shunt active filter with existing P&O MPPT	4.98
System with shunt active filter with PSO MPPT	1.81

5 Conclusions

The efficient operation of the proposed system for the power quality is verified through simulation with different loads. System behaviour is analysed before and after connecting filter and PV. PV fed shunt active power filter not only reduces the harmonics contents in the signals efficiently but also compensates the reactive power in an efficient manner. In PV system, MPPT algorithm is implemented to export the maximum active power to the load as and when required. PSO is verified as the efficient MPPT compared to the P&O MPPT. THD analysis of source current without PV gives good results but with PV, the performance is improved by reducing the harmonics contents which are within the IEEE 519 standard can be observed from Figs. 15 and 16.

References

1. Panchbhai A, Parmar S, Prajapati N (2017) Shunt active filter for harmonic and reactive power compensation using p-q theory. IEEE
2. Padamavati K, Sudha K (2019) Optimal sizing and placement of dynamic voltage restorer in the distribution system using firefly algorithm .Int J Energy Technol Policy 15:48–165
3. Zhou Y, Li H (2014) Analysis and suppression of leakage current in cascaded multilevel inverter based PV systems. IEEE Trans Power Electron 29:5265–5277
4. Balakrishna A, Babu M, Sreenivas L, Tulsi L (2012) Particle swarm optimization for power quality improvement based on shunt active power filter. Int J Sci Res 3(1):158–163
5. Abdul Kadir M, Yatim A, Yusuf S (2014) An improved PSO-based MPPT control strategy for photovoltaic systems. Int J Photo Energy
6. Puhan PS, Ray P, Panda G (2015) A comparative analysis of shunt active power filter and hybrid active power filter with different control technique applied for harmonic elimination in a single phase system. Int J Model Identif Control 24:19–28
7. Vavilapalli S, Padmanaban S, Subramaniam U, Mihet-Popa L (2017) Power balancing control for grid energy storage system in photovoltaic applications—real time digital simulation implementation. Energies 10:928
8. Baratar D, Buticchi G, Lorenzani E, Concari C (2014) Active common-mode filter for ground leakage current reduction in grid-connected PV converters operating with arbitrary power factor. IEEE Trans Ind Electron 61:3940–3950
9. Babes B, Rahmani L, Bouafassa A, Hamouda N (2005) Real time implementation of shunt active power filter for harmonic suppression and power quality improvement. IEEE Trans Ind Electron 52
10. Puhan PS. Ray P, Panda G (2018) A comparative analysis of artificial neural network and syn. detection controller to improve power quality in single phase system. Int J Power Electron 9:385–401
11. Djeghloud H, Bentounsi A, Benalla H (2011) Sub and super-synchronous wind turbine-doubly fed induction generator system implemented as an active power filter. Int J Power Electron 3(2)
12. Romero-Cadaval E, Spagnuolo G, Franquelo LG, Ramos-Paja C-A, Suntio T, Michael XW (2013) Grid-connected photovoltaic generation plants. IEEE Ind Electron 7(3):6–20
13. Akagi H, Kanazawa Y, Nabae A (2008) Instantaneous reactive power compensators comprising switching devices without energy storage components. IEEE Trans Ind Appl 3:625–630
14. Liu YH, Huang SC, Huang JW, Liang WC (2012) A particle swarm optimization-based maximum power point tracking algorithm for PV systems operating under partially shaded conditions. IEEE Trans Energy Convers 27

15. Singh M, Khadkikar V, Chandra A, Verma K (2011) Grid interconnection of renewable energy sources at the distribution level with power quality improvement features. IEEE Trans Power Deliver 26(1):307–15

Flood Forecasting Using Hybrid SVM-GOA Model: A Case Study

Abinash Sahoo and Dillip Kumar Ghose

Abstract A vital necessity in integrated water resources management is flood forecasting. Over the last few decades, artificial intelligence (AI) models have significantly contributed towards simulating physical flood processes by providing economical solutions and improved accuracy. In this study, a hybrid prediction model integrating support vector machine with grasshopper optimisation algorithm (SVM-GOA) is developed and evaluated for its suitability for flood prediction considering metrological data from Monierkhal station, Assam, India. Estimates provided by SVM-GOA model are compared with the standalone SVM model based on root mean square error (RMSE) and Willmott's Index (WI). Rainfall, water level, and discharge are important variables for flood estimation and are used as input variables in the present study. The best performance was found for SVM-GOA model with RMSE—0.0358, and WI—0.9838. Therefore, results specified that developed SVM-GOA model can be selected as an efficient technique over SVM for accurate flood prediction.

Keywords Flood · Monierkhal · SVM-GOA

1 Introduction

Forecasting flood events is an essential constituent of both flood warning and management. Not only it is an efficient tool for reducing several threats posed by floods on life, infrastructures, and property, but it can also deliver vital information to decision-makers for proper water resource management [1–4]. Recently, flood damage has become even more aggravated by rapid economic progress [5]. Several models developed for flood forecasting mainly apply three approaches in the last decades: statistical, conceptual, or AI. Even though the conceptual process has consistent accuracy and an apparent physical concept, huge data necessity limits its usage.

A. Sahoo (✉) · D. K. Ghose
Department of Civil Engineering, National Institute of Technology Silchar, Assam, India
e-mail: bablusahoo1992@gmail.com

In contrast, the statistical process is simple in formulating and executing. However, it does not give acceptable results as it is static and lacking in revealing a nonlinear relationship between specified variables. Compared to conceptual and statistical approaches, models based on AI, like ANNs, can learn fluctuating relationships amid input and output without knowing the system's physical processes. There have been many applications of statistical, ANN, adaptive neuro-fuzzy inference system (ANFIS), and SVM models in flood forecasting studies [6–10]. Major reason behind application of SVM model in the present study includes: with an appropriate kernel function, any complex problem can be solved, and SVM has a good generalisation ability compared to other neural network models. A new flood forecasting model was established for Wangjiaba station at Huaihe River, China based on boosting learning algorithms and SVM [11]. Outcomes revealed that proposed SVM ensemble model effectively improved flood forecasting accurateness. SVM model was applied for forecasting flash flood in a trivial mountainous watershed at various lead times [12]. ANN, RNN (recurrent neural network), and LSTM (Long Short-Term Memory) models were proposed for flood forecasting in Da River basin, Vietnam [13]. They found that LSTM model performed better in flood forecasting at proposed location. Deep neural network (DNN) was employed for predicting occurrence of flood based on rainfall intensity and temperature and compared the obtained results with SVM, K-nearest neighbour (KNN), and Naïve Bayes [14]. Results indicated that DNN efficiently predicted flood occurrence with highest accuracy.

But, their incapability in producing an obvious model for application by other investigators is a clear shortcoming. Moreover, their network solutions are easily stuck in local optima, and optimal structures are difficult to find. To overcome these shortcomings, optimisation algorithms are applied. Performance of KNN, radial basis neural network (RBFN), feed-forward neural network (FFNN), RNN, time-delay neural network (TDNN), and SVM-GOA models were investigated for monthly flow prediction [15]. ELM-PSO (extreme learning machine with particle swarm optimisation) was proposed for flood forecasting at Jenapur gauging station of Brahmani River, Odisha [16]. They concluded that proposed model showed promising results with more reliability and accuracy. PSO, genetic algorithm (GA), krill algorithm (KA), weed algorithm (WA), cat swarm optimisation (CSO), and GOA algorithms were used to improve performance of ANFIS, SVM, and ANN models in predicting monthly groundwater level [17]. Results revealed that ANFIS-GOA performed superiorly to other applied models for GWL prediction.

The main contribution of this paper is to assess performance of SVM-GOA model for monthly flood prediction considering historical data from Monierkhal station. Discharge, water level, and rainfall data of the study area are considered as input data of proposed model. Application of SVM-GOA in flood forecasting is the novel aspect of present study.

2 Study Area

Barak River basin lies between 22° 44′ N to 25° 58′ N and 89°50′ E to 94° 0′ E and covers almost 1.38% of total geographical area. It is the second largest river in northeast part of India. Tributaries of this river are mostly rainfed, causing floods when rain occurs. Barak River travels nearly 102 km in a zigzag way before reaching Bangladesh by flowing through alluvial plains of Assam. For present study, data from Monierkhal gauging station located in Cachar district of Assam is collected (Fig. 1).

3 Methodology

3.1 Support Vector Machine

In machine learning, SVM is a well-known powerful tool proposed by Cortes and Vapnik [18] to solve classification issues. A notable advantage of SVM is that it works based on structural risk minimisation, which optimises target function while considering complexity of the structure. It is well known that with the help of kernels, SVM functions. Among various kernel functions, RBF has proved to be the best and has further been investigated in hydrological studies, integrated with a linear function [19]. The decision function of the SVR model can be written as:

$$f(x) = w \cdot \varnothing(x) + b \qquad (1)$$

Here x—input vector; $\varnothing(x)$—nonlinear mapping function; b and w—constraint vectors of function. Now, aim of SVM is to discover optimal w, b and certain internal constraints of $\varnothing(x)$. By solving the subsequent optimisation problem, optimum solution of SVM can be found:

$$\min_{w,b,\xi,\xi^*} \frac{1}{2} w^2 + C \sum_{i=1}^{n} (\xi_i + \xi_i^*) \qquad (2)$$

where C—coefficient of positive constant penalty determining penalised loss degree when a training error takes place and ξ and ξ_i^*—slack variables (Fig. 2).

3.2 Grasshopper Optimisation Algorithm

GOA algorithm mimicking swarming behaviour of grasshoppers in nature was proposed [20]. A grasshopper's flying route in a swarm is influenced by three
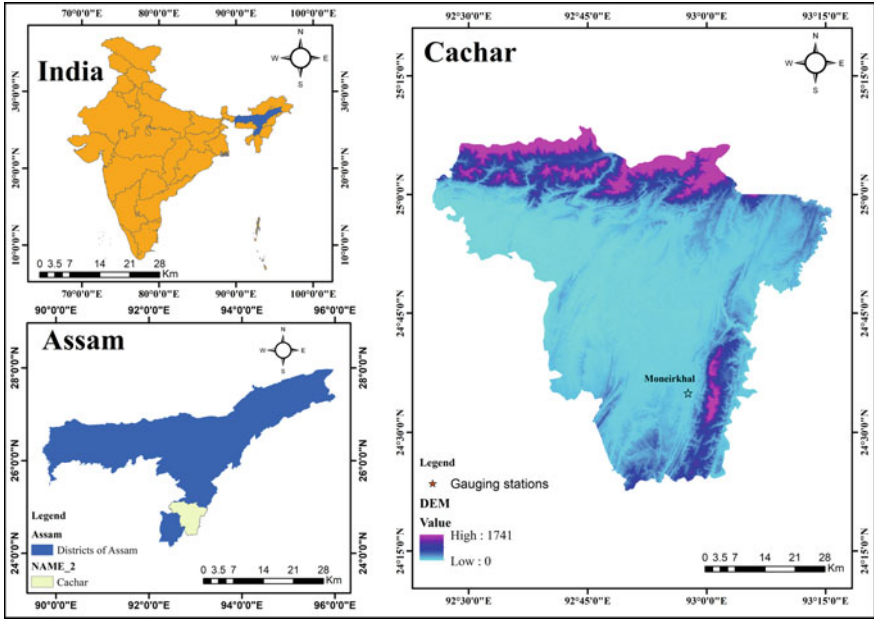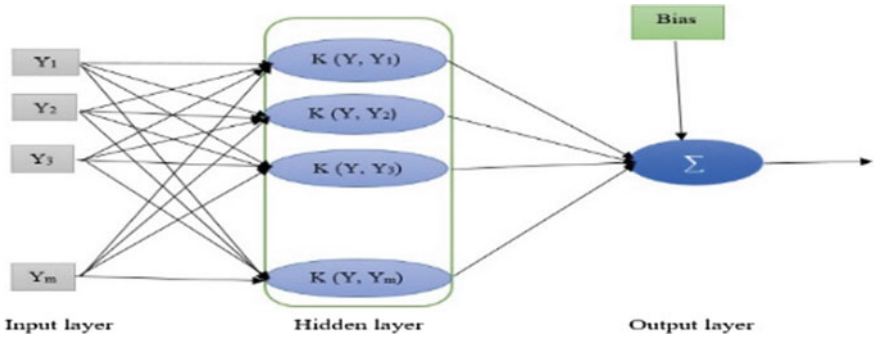
Fig. 1 Location of selected rain-gauge station

Fig. 2 Architecture of SVM network

constituents: gravity (G_i), wind advection (A_i), and social interaction (S_i). In GOA, the main mechanism of search is social interaction. It can be mathematically expressed as:

$$S_i = \sum_{j=1, j \neq 1}^{N} s(d_{ij}) \widehat{d_{ij}} \tag{3}$$

where d_{ij}—distance between ith and jth grasshopper, and it is calculated as $d_{ij} = |x_j - x_i|$, s—function defining power of societal forces, and $\widehat{d_{ij}} = \frac{x_j - x_i}{d_{ij}}$—unit vector from ith grasshopper to jth grasshopper.

From Eq. 3, it may be observed that function s is the key constituent of social interface. 's' describes grasshopper's movement direction in swarm and is expressed by:

$$s(r) = f e^{\frac{-r}{l}} - e^{-r} \tag{4}$$

where f—attraction intensity and l—attractive length scale. Interactions between grasshoppers are efficiently simulated by swarm model. But, it must be accustomed to designing an optimisation algorithm. Following equation represents the mathematical model:

$$x_i^d = c \left(\sum_{j=1, j \neq 1}^{N} c \frac{ub_d - lb_d}{s} s \left(|x_j^d - x_i^d| \right) \frac{x_j - x_i}{d_{ij}} \right) + \widehat{T_d} \tag{5}$$

where ub_d and lb_d—upper and lower bounds in dth dimension respectively, $\widehat{T_d}$—dth dimension's value in target (optimal solution obtained up to now), and c—decreasing coefficient for shrinking comfort area, attraction area, and repulsion area. The main controlling parameter in GOA algorithm is parameter c and, which is updated utilising subsequent expression:

$$c = c_{max} - l \frac{c_{max} - c_{min}}{L} \tag{6}$$

where L—maximum number of iterations, l—present iteration, $c_{max} = 1$, and $c_{min} = 0.00002$. SVM-GOA flowchart is given in Fig. 3.

3.3 Evaluating Constraint

Discharge (D_t), water level (S_t), and Rainfall (P_t) data of 50 years (1971–2020) were collected from IMD Pune. 70% of collected data, i.e. 1971–2005, were used for training while remaining 30%, i.e. 2006–2020, for testing the proposed models. Following statistical measures are considered for assessing accurateness of predictive models:

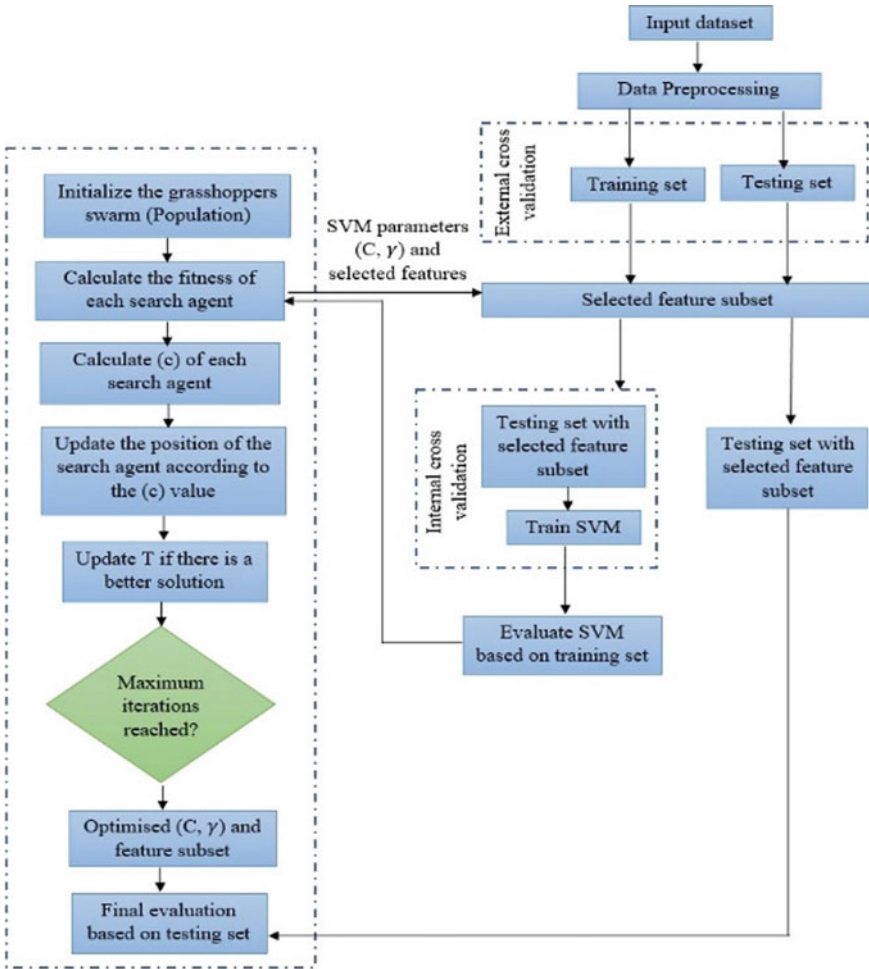$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (P_i - O_i)^2} \tag{7}$$

Fig. 3 Flowchart of SVM-GOA algorithm

$$R^2 = \left(\frac{\sum_{i=1}^{n} P_i - \overline{P_i})(O_i - \overline{O_i})}{\sqrt{\sum_{i=1}^{n} P_i - \overline{P_i})^2 (O_i - \overline{O_i})^2}} \right)^2 \tag{8}$$

$$\mathrm{WI} = 1 - \left[\frac{\sum_{k=1}^{N} (O_i - P_i)^2}{\sum_{k=1}^{N} \left(\left| P_i - \overline{O_i} \right| + \left| O_i - \overline{O_i} \right| \right)^2} \right] \tag{9}$$

where P_i = Estimated value; O_i = Measured value; $\overline{P_i}$ = Mean estimated value; $\overline{O_i}$ = Mean measured value. For best performance of the model, value of RMSE must be smaller and R^2 and WI must be higher.

4 Results and Discussions

In present study, a simple and efficient model is proposed for flood forecasting, and features of input datasets are also considered since it affects the performance of the models. Such features comprise quantity and type of input data series. Input scenarios with different input combinations are presented in Table 1. Table 2 presents performance of applied models, which depicts that SVM-GOAIII model with R^2—0.9813, RMSE—0.0358 and WI—0.9838 in testing phase are superior to conventional SVM models. This is due to the fact that SVM-GOA model apprehends short, intermediate, and long patterns in flow time series.

Scatter plots of observed versus predicted flood discharge for SVM and SVM-GOA prediction models during testing phase are shown in Fig. 4. It also approves that SVM-GOA forecasting model has best performance based on coefficient of determination (R^2) value.

Figure 4 gives information concerning graphical performance measures when data sets are near to 45° line. Data sets closest to 45° line signify better prediction results.

A clear graphical illustration of how SVM and SVM-GOA models have forecasted flood magnitude in training and testing phases is presented in Fig. 5. While SVM-GOA model predicts maximum flood magnitude of 4736.63 (m³/s) against actual flood value of 4879.61 (m³/s), SVM forecasts flood value of 4593.17 (m³/s). Moreover, it is evident that forecasting peak values of flood, SVM-GOA model had a greater ability than SVM and could give a more precise flood estimation.

A supplementary assessment has been performed for further evaluating performance of SVM-GOA model, based on observed and predicted flood values utilising

Table 1 Modelling input combination structure

Model description		Output variable	Input combinations
SVM	SVM-GOA		
SVMI	SVM-GOAI	F_t	D_t
SVMII	SVM-GOAII	F_t	D_t, S_t
SVMIII	SVM-GOAIII	F_t	D_t, S_t, P_t

Table 2 Performance of model

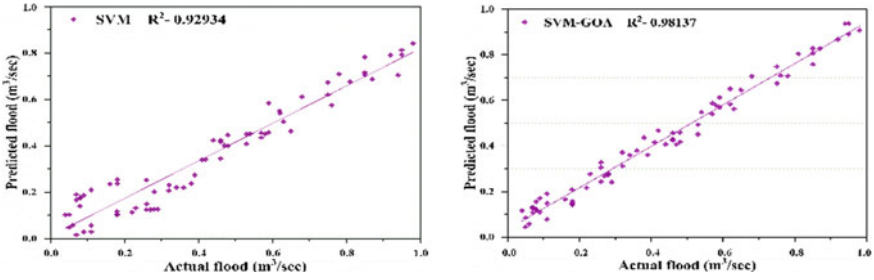| Technique	Input	R^2	RMSE	WI	R^2	RMSE	WI
		Training phase			Testing phase		
SVM	SVMI	0.9279	0.0812	0.9341	0.9148	0.0864	0.9306
	SVMII	0.9314	0.0637	0.9396	0.9235	0.0782	0.9352
	SVMIII	0.9386	0.0564	0.9472	0.9293	0.0631	0.9425
SVM-GOA	SVM-GOAI	0.9753	0.0502	0.9821	0.9627	0.0549	0.9809
	SVM-GOAII	0.9824	0.0391	0.9847	0.9749	0.0473	0.9737
	SVM-GOAIII	0.9875	0.0285	0.9896	0.9813	0.0358	0.9838

Fig. 4 Scatter plot of observed versus forecasted flood during testing phase
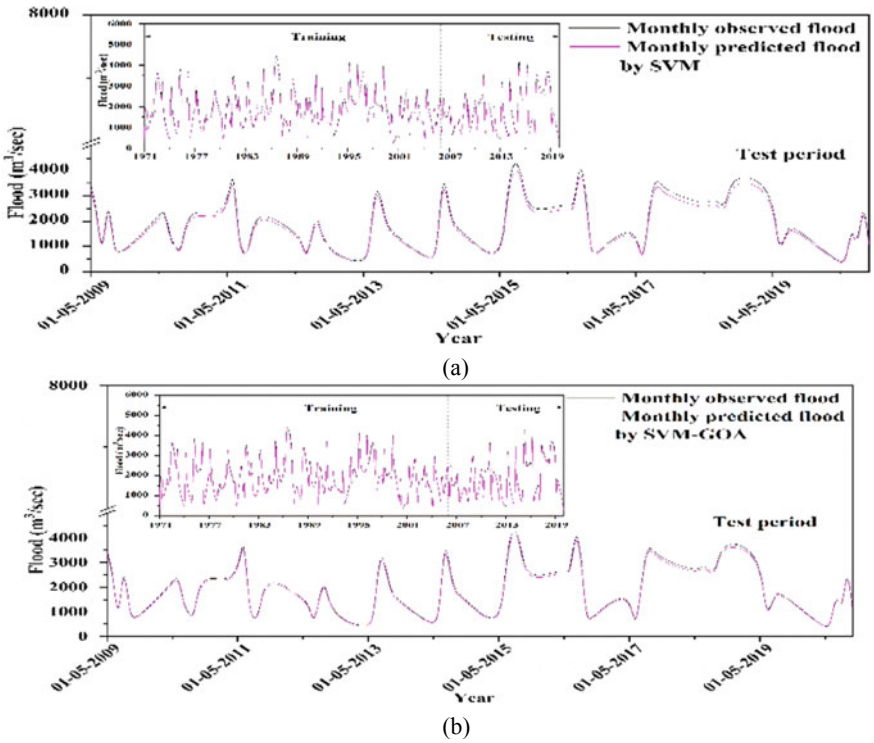

(a)

(b)

Fig. 5 Time series plot depicting observed versus predicted flood values at Monierkhal station

probability distribution of data in form of a histogram plot (Fig. 6). In summary, all results obtained above from statistical and graphical representations specify that SVM-GOA model is a powerful tool for modelling monthly flood during monsoon season and can yield more consistent performance than SVM. It gives an effective alternate for flood forecasting.
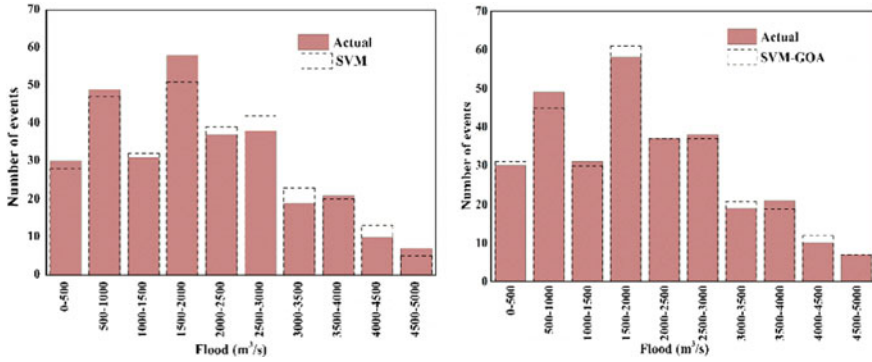
Fig. 6 Histogram plots of proposed models for flood forecasting

5 Conclusion and Future Scope

In this study, hybrid SVM-GOA model was developed and evaluated for flood prediction using data from Monierkhal station in Cachar district of Assam, India. SVM-GOA was also compared with a classical SVM model. Results indicated better performance of hybrid SVM-GOA model based on values of R^2, RMSE and WI. Therefore, it can be concluded that GOA is a powerful add-on tool for increasing forecasting preciseness of standalone SVM technique applied in flood estimation. Future study would include exploring the usability of model with more hydro-meteorological inputs for determining their impact on accurateness of flood estimation and also involve developing hybridised models that can be utilised for working with prevailing hydrological models.

References

1. Chen L, Singh VP, Lu W, Zhang J, Zhou J, Guo S (2016) Streamflow forecast uncertainty evolution and its effect on real-time reservoir operation. J Hydrol 540:712–726
2. Huang K, Ye L, Chen L, Wang Q, Dai L, Zhou J, Singh VP, Huang M, Zhang J (2018) Risk analysis of flood control reservoir operation considering multiple uncertainties. J Hydrol 565:672–684
3. Samantaray S, Sahoo A (2021) Estimation of flood frequency using statistical method: Mahanadi River basin, India. H$_2$Open J 3(1):189–207
4. Sahoo A, Samantaray S, Bankuru S, Ghose DK (2020) Prediction of flood using adaptive neuro-fuzzy inference systems: a case study. In: Smart intelligent computing and applications. SIST, vol 159, pp 733–739. Springer, Singapore
5. Mikhailov VN, Morozov VN, Cheroy NI, Mikhailova MV (2008) Extreme flood on the Danube River in 2006. Russ Meteorol Hydrol 33:48–54
6. Sahoo A, Ghose DK (2021) Flood frequency analysis for menace gauging station of Mahanadi River, India. J Inst Eng (India): Ser A 1–12
7. Samantaray S, Sahoo A, Agnihotri A (2021) Assessment of flood frequency using statistical and hybrid neural network method: Mahanadi River Basin, India. J Geol Soc India 97(8):867–880

8. Sahoo A, Singh UK, Kumar MH, Samantaray S (2021) Estimation of flood in a river basin through neural networks: a case study. In: Communication software and networks LNNS, vol 134, pp 755–763. Springer, Singapore

9. Sahoo A, Samantaray S, Paul S (2021) Efficacy of ANFIS-GOA technique in flood prediction: a case study of Mahanadi river basin in India. H_2Open J 4(1):137–156

10. Sahoo A, Samantaray S, Ghose DK (2021) Prediction of flood in Barak river using hybrid machine learning approaches: a case study. J Geol Soc India 97(2):186–198

11. Li S, Ma K, Jin Z, Zhu Y (2016) A new flood forecasting model based on SVM and boosting learning algorithms. In: 2016 IEEE Congress on evolutionary computation (CEC), pp 1343–1348. IEEE

12. Wu J, Liu H, Wei G, Song T, Zhang C, Zhou H (2019) Flash flood forecasting using support vector regression model in a small mountainous catchment. Water 11(7):1327

13. Le XH, Ho HV, Lee G, Jung S (2019) Application of long short-term memory (LSTM) neural network for flood forecasting. Water 11(7):1387

14. Sankaranarayanan S, Prabhakar M, Satish S, Jain P, Ramprasad A, Krishnan A (2020) Flood prediction based on weather parameters using deep learning. J Water Clim Change 11(4):1766–1783

15. Alizadeh Z, Yazdi J, Kim JH, Al-Shamiri AK (2018) Assessment of machine learning techniques for monthly flow prediction. Water 10(11):1676

16. Anupam S, Pani P (2020) Flood forecasting using a hybrid extreme learning machine-particle swarm optimization algorithm (ELM-PSO) model. Model Earth Syst Environ 6(1):341–347

17. Seifi A, Ehteram M, Singh VP, Mosavi A (2020) Modeling and uncertainty analysis of groundwater level using six evolutionary optimization algorithms hybridized with ANFIS, SVM, and ANN. Sustainability 12(10):4023

18. Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20(3):273–297

19. Dibike YB, Velickov S, Solomatine D, Abbott MB (2001) Model induction with support vector machines: introduction and applications. J Comput Civ Eng 15:208–216

20. Saremi S, Mirjalili S, Lewis A (2017) Grasshopper optimisation algorithm: theory and application. Adv Eng Softw 105:30–47

Fractal-Based Soil Assessment to Obtain Precision Agriculture Using Machine Learning Approach

Rajalaxmi Padhy, Shubhanshu Biswal, Sanjit Kumar Dash, and Jibitesh Mishra

Abstract Soil particle composition and its properties are important indicators used for tracking and evaluating the soil quality and fertility of soil which is an important aspect in agriculture. Soil fractal dimension is an intrinsic property of soil and represents a comprehensive index of soil physio-chemical properties. Fractal concepts can be implemented in soil technology to research and quantify its role in agriculture. In this paper, robust machine learning models are used to find the relation and importance of soil fractal dimension in agriculture. Linear regression model is used to predict the relationship between fractal dimension of soil and different soil properties. Soil quality is calculated for every data point to find the relationship between soil quality and soil fractal dimension. To achieve this, another linear regression model is used to predict the relationship between fractal dimension and quality of soil. The plot was found as a result suggested that soil quality is directly related to soil fractal dimension. Crop analysis is then performed to find the influence of soil fractals on agriculture using another linear regression model. The result gave an overview of the relationship and importance of soil fractals on agriculture. The change in fractal dimension of soil is found to be directly related to change in soil properties, and thus, soil fractal dimension is directly related to agriculture. Thus, it laid a theoretical reason for more investigations on soil fractals which are important for attaining precision agriculture.

Keywords Fractal dimension · Mass volume fractal model theory · Physio-chemical properties · Soil fractal dimension · Regression analysis

1 Introduction

Soils are intricate combinations of minerals, water, air, natural matter and endless creatures. It is made out of soil debris of different arrangements and measurements [1, 2]. Soil constructions and its structures are calculated by means of soil molecule

---

R. Padhy · S. Biswal · S. K. Dash (✉) · J. Mishra
Odisha University of Technology and Research, Bhubaneswar, Odisha, India
e-mail: sanjitkumar303@gmail.com

length dispersion, which randomly impacts soil dampness attributes, ripeness and soil disintegration [3]. Eventually, many soil researchers have utilized the soil's physical properties to find actual properties of water maintenance, mass thickness, porousness, etc. [4, 5]. Till a definite amount, soil physical properties are imperative records for the assessment of soil and its relationship with different soil features [6–9]. In regions with high soil deterioration, costs as a result of precipitation and overflow, soil physical properties (noticed by means of nutrients) are explicitly taken out or put away at some stage in soil disintegration procedures [10–15]. The soil fractal designs of different lands could be different as shown in Fig. 1.

Taking all the factors into account, the research work has been designed accordingly to meet the objectives. The research work mostly lies around soil fractal dimension, soil properties and agricultural aspects which are related to soil characteristics. The objective of the paper is to describe and assess the status of the fractal dimensions of different types of soil and investigate the relationship between soil fractal dimensions and soil properties. Assess the status of soil quality of different soil types and investigate the relationship between soil fractal dimensions and soil quality. Explore the possibility of influence of the fractal dimension of soil in agriculture. The rest of the paper is organized as follows: Sect. 2 presents the review of the existing literature. Section 3 discusses the system model design used for designing the flow of procedure. Section 4 explains the dataset analysis. Section 5 includes the results and discussions which help in quantifying the overall result. Section 6 concludes the paper and discusses future scope of the research work.
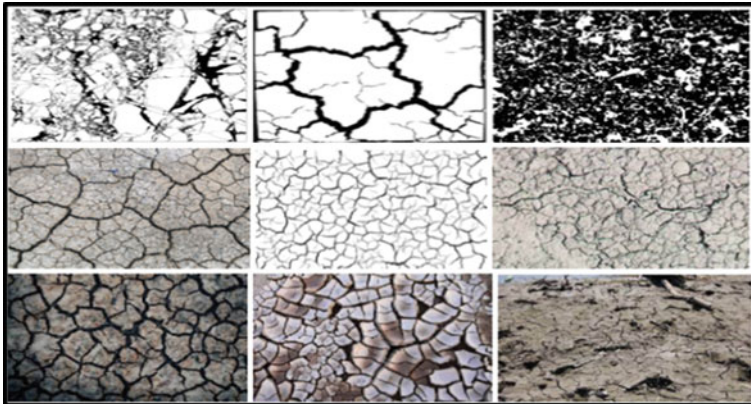

Fig. 1 Soil fractal imaging of different regions

2   Literature Review

Previous work in this area includes methods to find fractal measurements that rely on soil mass fractal analysis, soil volume fractal analysis and other methods based on the physio-chemical properties to get the overview of the relationship of soil mass fractal measurement with different soil properties. Many researchers have given their contribution in this field, from which we have taken inspiration and proposed our novel approach for finding the relationship of soil fractal dimension with different soil properties and depict an overall influence of soil fractal dimension on quality and fertility of soil and thus on agriculture. Tyler and wheatcraft [4]: developed a soil mass distribution technique to assess the shape estimation of physical properties of soil and progressed the boundaries of shape conduct and applications for soil physical properties. The outcomes of their result helped in understanding the core knowledge about the soil particles and how their behaviour changes with the change in soil fractals. This study helped us understand that soil fractals have a relationship with soil particles distributions which formed the base of our study. Yang et al. [5]: executed dead shape scaling normal for the depiction of soil physical characteristics of assorted soil surfaces with reference to the soil mass shape analysis technique. The outcomes in their examinations established that shape-size investigation modified from sand to clay content. The study also helped in understanding the behaviour of soil content and how soil particles content changes with the change in fractal pattern of the soil. Huang and Zhan [6]: got conclusive evidence that if the clay content will increase or decrease, shape measurements conjointly will increase or decrease consequently; however, if the sand content will increase, shape measurements decrease and vice versa.

3   System Model

In this section, we have proposed our methodology and approach which consists of three sections, i.e. data preprocessing, model training stage and result overview as shown in Fig. 2. The aim of the approach is to figure out the overall influence of soil fractals on agriculture. In the data preprocessing stage, the raw datasets were filtered and redundant variables and meta-data were removed which were unnecessary; afterwards, soil fractal dimension values were calculated and matched with their respective data points. In the model training stage, the datasets were trained using linear regression analysis to depict the relationship status between the soil properties and soil fractal dimension. Afterwards, soil quality analysis was done whose results were used in training the second model to find the relationship of soil fractal dimension with soil quality. We then performed the crop analysis and with the help of the previous results, we trained the third model with linear regression again to depict the influence of soil fractal dimension on agriculture. The results of these models, soil quality analysis and crop analysis are discussed in the result section.
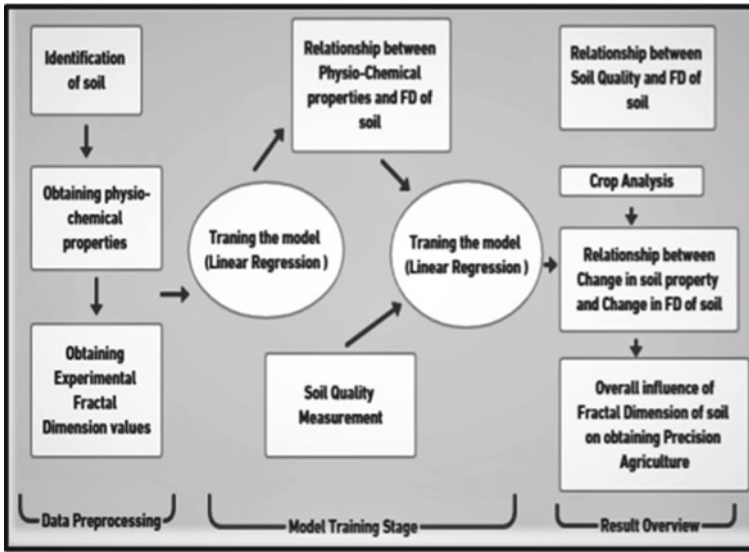
Fig. 2 Overview of the model architecture

3.1 Data Preprocessing and Training

The soil testing has been done on the three important soil types for agriculture: (*alluvial, black, red and yellow*) as shown in Fig. 3. The average error percentage added in the dataset was removed to perfectly fit into the model.

We divided the entire dataset into smaller datasets, which describe the relation of each soil property with fractal dimension. For choosing the best model for the approach, we took two models into consideration, i.e. linear and nonlinear regression. The nonlinear regression would have generated complex quadratic relations between the two dependent variables, and there could have been more than one parameter to analyse the soil properties, but we wanted a simple linear relation between the two dependent variables so that every soil property could be measured with a single parameter, i.e. either direct or indirect. For this reason, we performed linear regression to train the models.

3.2 Soil Quality Measurement

For finding the influence of physical and chemical properties on soil quality, we referred to the study [12, 16], where the method of multivariate principal component analysis (PCA) was used to reduce the size of independent variables and find the most important attributes according to their influence. We standardized the dataset and performed PCA for the removal of redundant variables, and we were left with
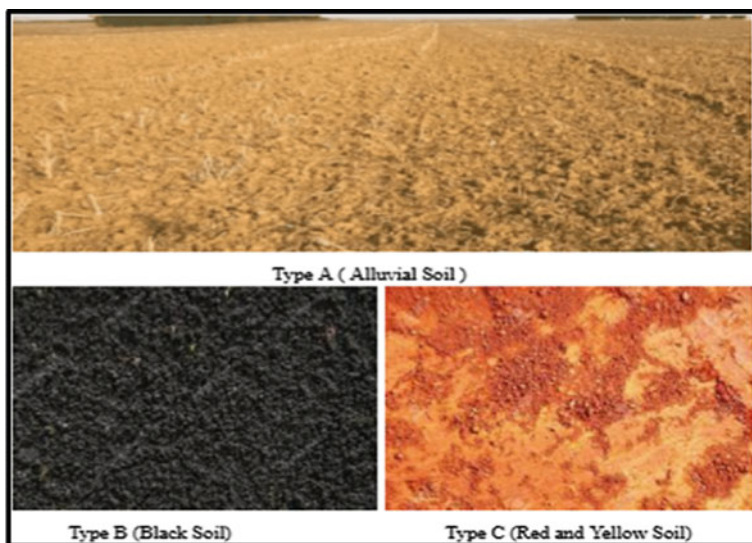
Fig. 3 Three different types of soil

five attributes instead of eleven and out of which all five were chemical properties. It was obtained from the results that chemical properties have more influence on soil quality than the physical properties and the order of influencing parameter was obtained as:

Soil Organic Carbon > Bulk density > Saturated Water Content
= pH Value > Total Porosity

Soil organic carbon (SOC) was found out to be the most influencing parameter while the total porosity was found out to be the least. According to the PCA analysis, the coefficients of the variables were found out and then combined together to form the formula for finding standardized soil quality as shown in Eqs. (1) and (2).

Simplifying Eq. (1) gives Eq. (2), from which soil quality can be calculated.

$$
\begin{aligned}
\text{Soil Quality} = [&0.39001 * (\text{Soil Organic Carbon}) \\
&+ 0.28002 * (\text{Bulk Density}) + 0.17001 * (\text{pH Value}) + 0.17001 \\
&* (\text{Saturated Water Content}) + 0.13002 * (\text{Total Porosity})]/1.140011
\end{aligned} \tag{1}
$$

$$
\begin{aligned}
\text{Soil Quality} = &0.340012 * (\text{Soil Organic Carbon}) \\
&+ 0.250001 * (\text{Bulk Density}) + 0.150001 * (\text{pHValue}) + 0.150001 \\
&* (\text{Saturated Water Content}) + 0.110002 * (\text{Total Porosity})
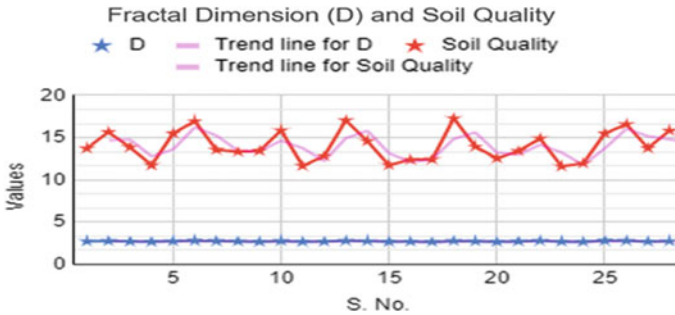\end{aligned} \tag{2}
$$

Fig. 4 Line graph representation of fractal dimension and soil quality trend

We then calculated soil quality of each datapoint with the help of Eq. 2. The line graph trend change of soil quality and fractal dimension is shown in Fig. 4, where the fractal dimension trend changes from 2.639 to 2.779 and soil quality trend changes from 11.5704 to 17.2808. The average of soil quality was found out to be 14.046725. The trailing average in trend was calculated using period as 2. The average change in soil quality was found out to be more than that of fractal dimension. Afterwards, with the data of soil quality and fractal dimension, we performed linear regression to find the relation of soil fractal dimension with soil quality.

4 Dataset Analysis

The dataset used in this paper is taken from various sources, i.e. soil texture fractions in 'dergipark.org', different soil particle size classification in 'mpdi.com', soil properties in 'plos.org' and fractal dream dataset in 'kaggle'. The original dataset has extra parameters like soil texture and error percentage added with each data point which was not required for this experiment, so these meta-data were removed from the dataset to ensure that such meta-data does not affect the training model. The complete dataset was further divided into two sub-datasets:

I. **According to physical properties of soil**
   The entire dataset has 7 columns and 29 rows where the 6 parameters have been considered into effect, i.e. gravel content (%), coarse sand content (%), fine sand content (%), silt content (%), soil fractal dimension. The soil types are divided into three types, i.e. Type A (Alluvial), Type B (Black), Type C (Red and Yellow) as shown in Table 1.

II. **According to chemical properties of soil**
   The entire dataset has 8 columns and 29 rows where the 7 parameters have been considered into effect, i.e. soil organic carbon (g kg^{-1}), bulk density (g cm^{-3}), saturated water content (%), total porosity (%), saturated hydraulic conductivity (mm min^{-1}), pH value, soil fractal dimension. The soil types are divided into

Table 1 Different physical properties with soil fractal dimension

Soil	Clay	Coarse sand	Fine sand	Slit	Gravel	D
Type A	13.13	22.8	23.03	23.02	18.02	2.708
Type A	16.54	22.53	15.12	26.46	19.35	2.748
Type A	10.84	24.56	16.76	28.62	19.22	2.687
Type A	9.67	26.12	25.34	18.77	20.11	2.661
Type A	15.09	18.97	18.77	29.6	17.57	2.733
Type A	19.99	16.01	14.54	30.22	19.24	2.779
Type A	15.47	19.94	12.54	36.59	15.47	2.741
Type A	11.92	19.55	18.71	28.14	21.69	2.703
Type A	9.53	22.58	28.95	20.46	18.49	2.658
Type A	17.29	24.4	21.92	25.83	10.56	2.745
Type B	8.99	22.07	21.87	25.7	21.37	2.661
Type B	10.54	23.07	17.83	28.35	20.21	2.683
Type B	19.56	16.61	18.22	32.85	12.76	2.769
Type B	15.1	16	20.14	30.06	18.7	2.736
Type B	9.81	27.78	25.12	16.72	20.57	2.662
Type B	11.2	21.8	23.3	20.47	23.23	2.688
Type B	8.14	28.29	21.62	17.66	24.29	2.639
Type C	18.06	14.76	21.62	37.61	11.03	2.757
Type C	13.29	21.65	20.93	22.55	21.58	2.713
Type C	8.7	32.1	18.52	20.83	19.85	2.649
Type C	12.57	21.32	20.76	27.65	17.7	2.705
Type C	17.98	21.96	15.2	30.13	14.74	2.76
Type C	9.49	32.87	15.55	15.83	26.25	2.666
Type C	8.14	29.54	15.92	16.37	30.03	2.649
Type C	20.22	15.99	18.32	28.09	17.38	2.777
Type C	19.38	19.24	20.35	26.69	14.33	2.764
Type C	10.06	22.76	16.58	25.36	25.24	2.679
Type C	14.55	21.08	16.11	28.58	19.67	2.732

three types, i.e. Type A (Alluvial), Type B (Black), Type C (Red and Yellow) as shown in Table 2.

5 Result and Discussions

In this section, we have discussed the observations and the results obtained after the regression analysis. These results have thus helped in understanding the nature and relationship of the soil properties and soil fractal dimension with each other. The soil

Table 2 Different chemical properties of soil with soil fractal dimension

Soil	Carbon	Soil density	PH value	Water content	Total porosity	Conductivity	D
Type A	8.86	1.44	5.09	30.34	45.72	1.80	2.708
Type A	11.69	1.33	4.94	34.28	49.77	0.79	2.748
Type A	10.25	1.40	5.02	27.09	47.22	1.40	2.687
Type A	4.50	1.38	4.97	25.53	47.92	2.15	2.661
Type A	12.49	1.41	5.22	33.15	46.67	1.01	2.733
Type A	14.76	1.25	5.11	33.83	52.37	0.72	2.779
Type A	9.06	1.48	4.94	30.13	44.02	0.69	2.741
Type A	8.68	1.47	5.06	28.98	44.53	1.57	2.703
Type A	8.52	1.43	4.90	29.18	45.99	1.46	2.658
Type A	12.65	1.38	4.99	34.42	48.06	0.92	2.745
Type B	3.94	1.47	5.11	28.52	44.52	2.16	2.661
Type B	5.10	1.37	5.06	31.24	48.16	1.30	2.683
Type B	14.26	1.35	4.97	38.03	49.04	0.69	2.769
Type B	10.14	1.37	5.24	31.20	48.38	1.35	2.736
Type B	4.58	1.32	4.92	23.95	50.13	2.16	2.662
Type B	6.81	1.36	4.97	24.32	48.54	1.83	2.688
Type B	5.54	1.31	5.01	23.95	50.74	1.69	2.639
Type C	15.79	1.42	4.83	38.14	46.47	0.67	2.757
Type C	11.11	1.36	4.78	25.05	48.50	1.20	2.713
Type C	7.12	1.43	5.04	26.32	45.89	2.60	2.649
Type C	6.88	1.40	4.90	31.95	47.18	1.06	2.705
Type C	11.38	1.51	5.08	34.41	42.92	1.09	2.76
Type C	4.94	1.48	4.99	26.12	44.13	1.82	2.666
Type C	6.71	1.49	5.04	24.65	43.94	2.33	2.649
Type C	13.71	1.53	5.34	33.33	42.24	0.47	2.777
Type C	15.92	1.46	5.03	33.97	45.02	0.70	2.764
Type C	8.29	1.37	4.88	30.32	48.29	1.72	2.679
Type C	10.69	1.39	5.07	38.86	47.7	1.10	2.732

quality analysis and crop analysis results have helped in understanding the overall influence of soil fractal dimension on agriculture which is discussed below.

5.1 Relationship Analysis of Soil Fractal Dimension with Soil Physical Properties

After performing linear regression analysis between fractal dimension and physical properties of soil, we found out two trends, i.e. direct or indirect relation. We found out that the gravel content and coarse sand content have a strong indirect relationship with soil fractal dimension with R^2 score of 0.504 and 0.634 as shown in Figs. 5 and 6, respectively, while fine sand content has a partially strong indirect relationship with soil fractal dimension with a R^2 score of 0.217 as shown in Fig. 7. The results suggest that gravel content and coarse sand content have more indirect influence on fractal dimension than fine sand content.
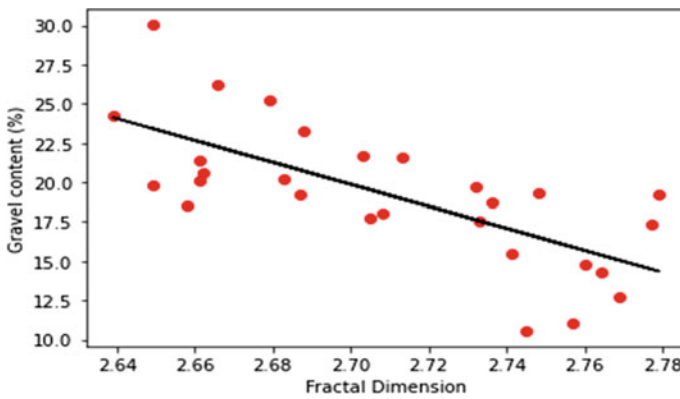
Fig. 5 Gravel content (%) versus soil fractal dimension
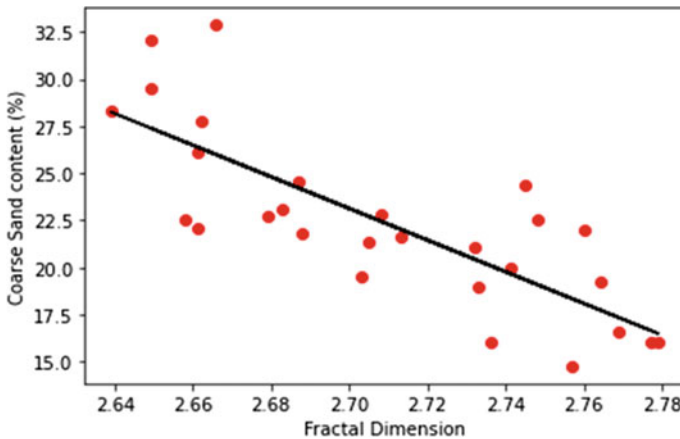
Fig. 6 Coarse sand content (%) versus soil fractal dimension
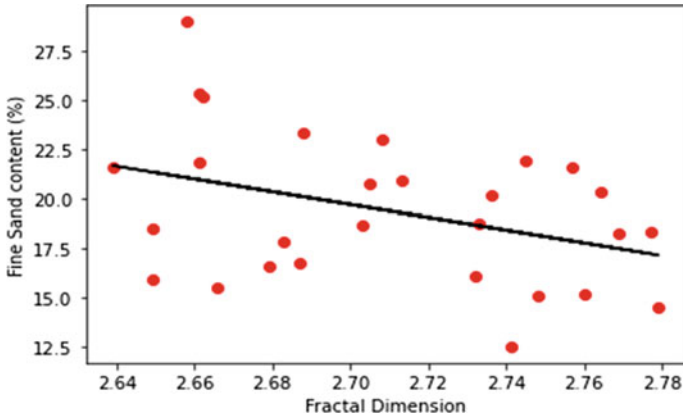
Fig. 7 Fine sand content (%) versus soil fractal dimension

We found that the clay content has a very strong direct relationship with soil fractal dimension with a R^2 score of 95.021 as shown in Fig. 8, while silt content has a partially strong direct relationship with soil fractal dimension with a R^2 score of 58.8 as shown in Fig. 9. The results suggest that clay content has more direct influence on fractal dimension than silt content.

The result suggests that the increase in quantity of clay and silt will increase the soil fractal dimension while the increase in quantity of gravel, fine sand and coarse sand will decrease the fractal dimension. We know that the clay content improves the soil texture while sand content reduces the quality of soil texture, so we can say that soil texture and its quality have a relationship with soil fractals and higher the fractal dimension better will be the soil texture and lower the fractal dimension,
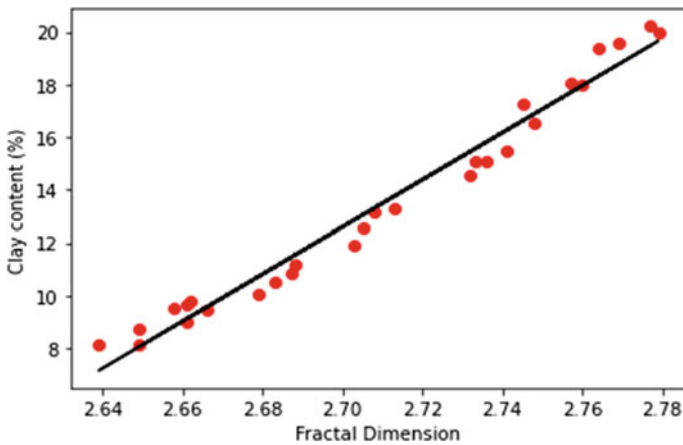
Fig. 8 Clay content (%) versus soil fractal dimension
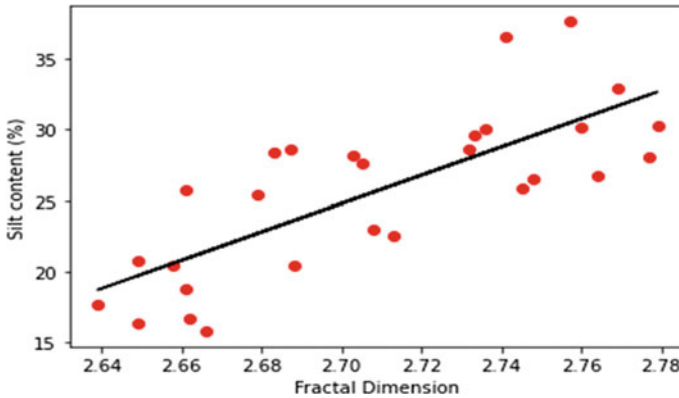
Fig. 9 Silt content (%) versus soil fractal dimension

worse will be the soil texture. Hence, soil texture improves with the increase in soil fractal dimension and soil productivity is directly linked with soil texture, so soil productivity will also increase with the increase in soil fractal dimension.

5.2 Relationship Analysis of Soil Fractal Dimension with Soil Chemical Properties

After performing linear regression analysis between fractal dimension and chemical properties of soil, we found out two trends, i.e. direct or indirect relation. We found out that the soil organic carbon content and saturated water content have a strong direct relationship with soil fractal dimension with a R^2 score of 0.787 and 0.649 as shown in Figs. 10 and 11, while the pH value has a partially strong direct relationship with soil fractal dimension with a R^2 score of 0.068. The results suggest that soil organic carbon content and saturated water content have more direct influence on fractal dimension than pH value (Fig. 12).

The saturated hydraulic conductivity has a strong indirect relationship with soil fractal dimension with a R^2 score of 0.798 as shown in Fig. 13. The result suggests that the saturated hydraulic conductivity has an indirect influence on the fractal dimension of soil. The bulk density and total porosity have neither strong direct or indirect relationship with soil fractal dimension as shown in Figs. 14 and 15, respectively. The result suggests that these properties have not a direct or indirect influence on soil fractal dimension.

The result suggests that the increase in soil organic carbon and saturated water content will increase the fractal dimension of soil while increase in soil hydraulic conductivity will decrease the soil fractal dimension. As the soil organic carbon is an organic matter and we know that organic matter and water content increase the soil quality and fertility, so we can conclude that increase or decrease in soil
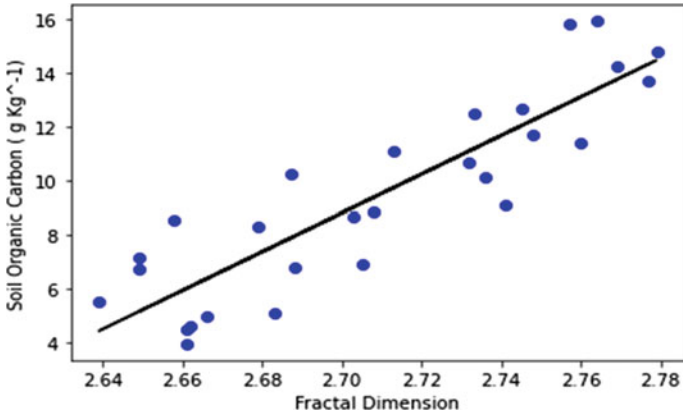
Fig. 10 Soil organic carbon (g kg^{-1}) versus soil fractal dimension
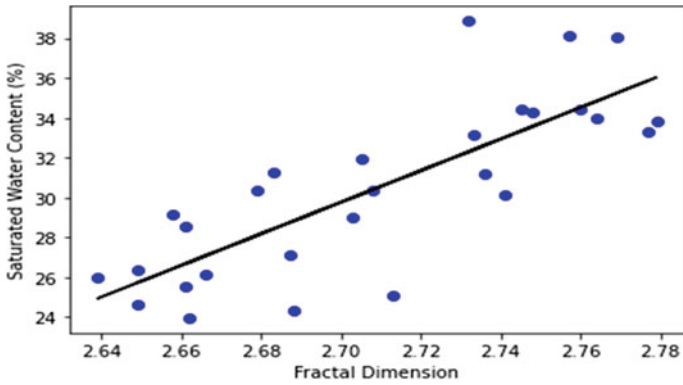

Fig. 11 Saturated water content (%) versus soil fractal dimension
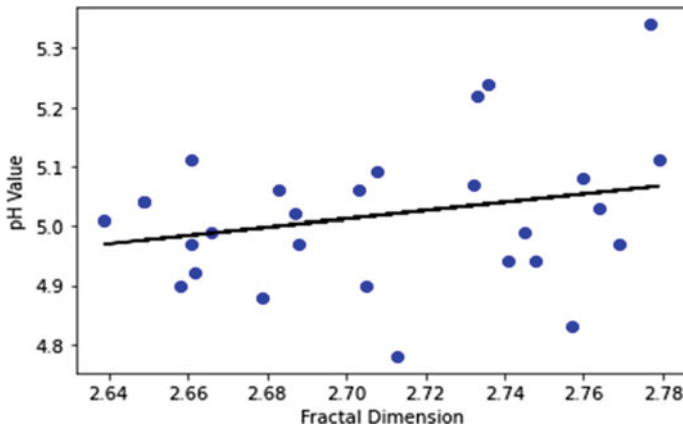

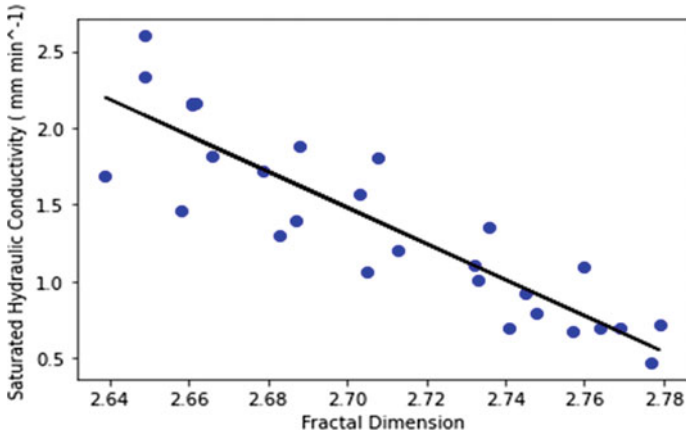**Fig. 12** pH value versus soil fractal dimension

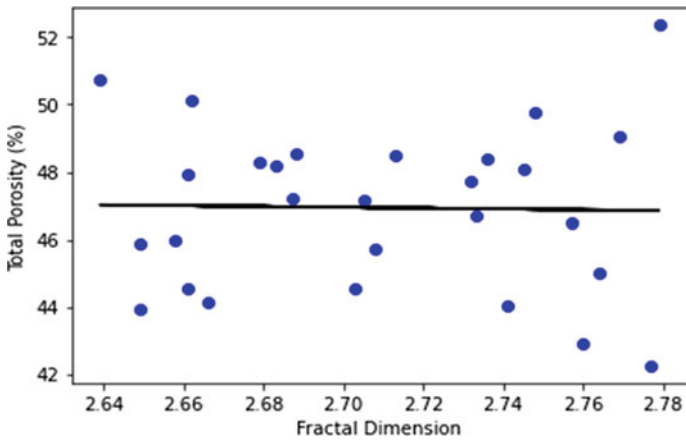**Fig. 13** Saturated hydraulic conductivity (mm min^{-1}) versus soil fractal dimension

Fig. 14 Total porosity (%) versus soil fractal dimension

fractal dimension will increase or decrease the soil fertility and vice versa. From both the results obtained (physical and chemical properties), we can say that soil fractal dimension has a strong relationship with the soil quality.

5.3 Relationship Analysis of Soil Fractal Dimension with Soil Quality

After performing soil quality analysis, we found out that soil quality is influenced by the physio-chemical properties and chemical properties have more influence on soil
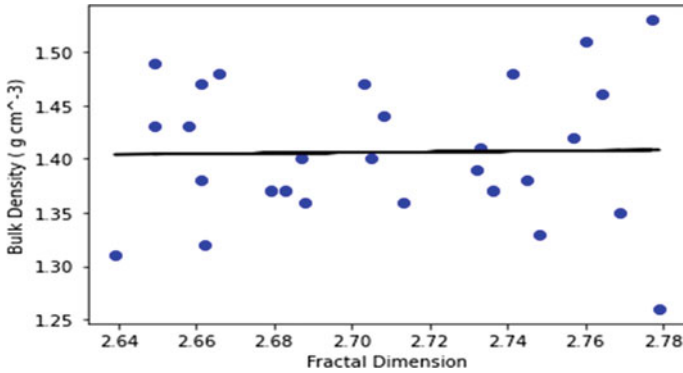
Fig. 15 Bulk density (g cm^{-3}) versus soil fractal dimension

quality than the physical properties. We also found that these properties have a link with soil fractal dimension. After performing linear regression analysis between soil quality and fractal dimension, we found that the fractal dimension of soil is directly related with the soil quality and has a very strong relationship with a R^2 score of 0.823 as shown in Fig. 15. Result suggests that the increase or decrease in fractal dimension of soil will affect the quality of soil accordingly and increase or decrease in soil quality will directly affect the soil fractal dimension. As agriculture is fully dependent on soil quality, we discovered that soil quality is dependent on fractal dimension, so we can say that agriculture is likewise dependent on fractal dimension of soil. That is, for acquiring precision agriculture, fractal dimension performs an essential function (Fig. 16).
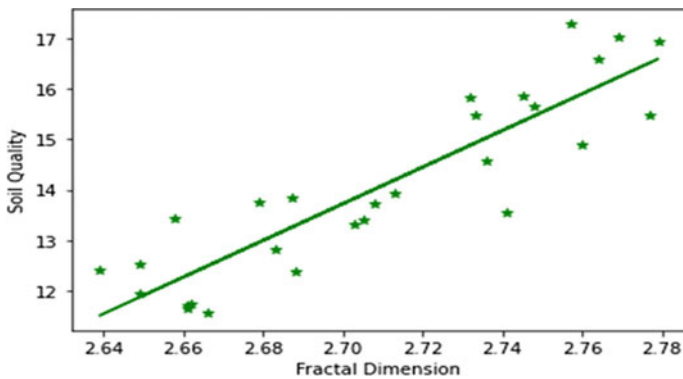

Fig. 16 Soil quality versus fractal dimension

5.4 Association of Fractal Dimension with Agriculture

Different types of soil have different chemical compositions and properties. Thus, a variety of crops grow on different soils. For agriculture, three main soil types are important: (i) alluvial soils (ii) black soils (iii) red and yellow soils.

So, we had performed this experiment considering these three soil types only.

Crops suitable for these soil types:

 (i)   **Alluvial soil**: Rice, wheat, sugarcane, tobacco, cotton, jute, maize, oilseeds, vegetables and fruits.
(ii)   **Black soil**: Cotton, wheat, jowar, linseed, virginia, tobacco, castar, sunflower, millets, rice, sugarcane, vegetables and fruits.
(iii) **Red and Yellow soil**: Cotton, wheat, rice, beans, millets, tobacco, pulses, oilseeds, potatoes and fruits.

Fractal dimension range of these soil types is:

Alluvial: 2.649–2.779, Black: 2.5831–2.8230, Red and Yellow: 2.693–2.713.

Fractal dimensions of these soil types are quite similar as we can see that the alluvial soil fractal dimension range lies within the black soil. Red and yellow soil fractal dimension range is also quite close to the black soil. It gives a very clear idea that fractal dimension of soil is closely related to the type of crop we want to grow.

As chemical properties of soil have more influence than its physical properties on soil quality. And SOC being the most contributing property, it was taken into consideration for the experiment. We then calculated the change in soil organic carbon and change in fractal dimension of each datapoint with the help of Table 2. The line graph trend of change in soil organic carbon and change in fractal dimension is shown in Fig. 17, where the change in fractal dimension trend changes from 0.013 to 0.128 and the change in soil organic carbon trend changes from 0.16 to 10.25. The average change in soil organic carbon was found out to be 4.2211 that can be seen as
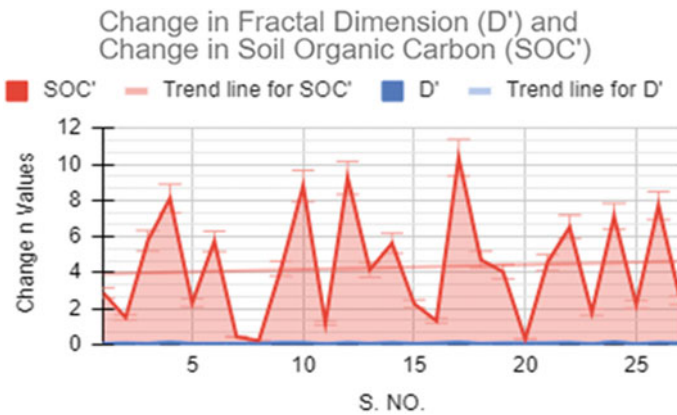

Fig. 17 Change in fractal dimension and change in soil organic carbon trend
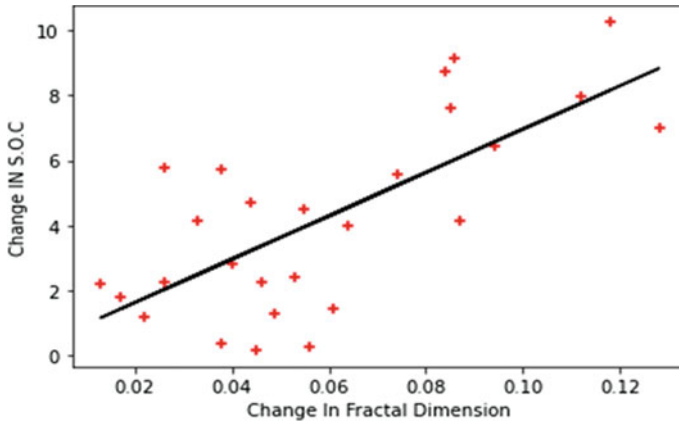
Fig. 18 Change in SOC versus change in fractal dimension

the trend line for soil quality lies around this value. The trailing average in trend was calculated using period as 2. The average change in soil organic carbon was found out to be more than that of change in fractal dimension.

Afterwards, regression analysis was performed to determine the relationship status between change in soil fractal dimension values and the change in chemical property (SOC). From the result, it was found that the change in soil fractal dimension has a direct and strong relationship with the change in SOC property with a R^2 score of 0.678 as shown in Fig. 18. With the result obtained from the experiment, we can say that the relation the soil property holds with the soil fractal dimension, whether direct or indirect, the change in that property will bring out the same change in the fractal dimension or vice versa accordingly. It suggests that the change in physio-chemical properties of the soil could also change the soil fractal dimension and vice versa. So, if a particular type of crop is to be grown on a particular soil, its fractal dimension could be shifted accordingly by changing the chemical or physical composition of the soil. As chemical properties have more influence on soil quality, and out of which soil organic property is most influencing and the other properties follow as such; accordingly, their quantity will differ in bringing a similar amount of change in fractal dimension. The results we got will help in understanding the impact and influence of soil fractals on agriculture and will help in obtaining precision agriculture.

6 Conclusion

The different soil types showed a huge impact on soil properties. If the soil texture improves the soil fractal dimension also improves. The increase in quantity of clay and silt will increase the soil fractal dimension while the increase in quantity of gravel, fine sand and coarse sand will decrease the fractal dimension. As the clay

content improves the soil texture while sand reduces the quality of soil texture, it can be said that higher the fractal dimension higher will be the soil texture quality and higher will be its productivity. The increase in soil organic carbon and saturated water content will increase the fractal dimension of soil while increase in soil hydraulic conductivity will decrease the soil fractal dimension. The increase or decrease in soil fractal dimension will increase or decrease the soil fertility and vice versa. Soil chemical properties have more influence on soil quality than the physical properties. Soil organic carbon was found out to be the most influencing parameter while the total porosity is the least influencing parameter in soil quality. The increase or decrease in fractal dimension of soil will increase or decrease the quality of soil, respectively, and vice versa. The relation the soil property holds with the soil fractal dimension, whether direct or indirect, the change in that property will bring out the same change in the fractal dimension or vice versa accordingly. It suggests that the change in physio-chemical properties of the soil could also change the soil fractal dimension and vice versa. So, if a particular type of crop is to be grown on a particular soil, its fractal dimension could be shifted accordingly by changing the chemical or physical composition of the soil. The results we got will help in understanding the impact and influence of soil fractals on agriculture and will help in obtaining precision agriculture.

References

1. Ding F, Huang Y, Sun W, Jiang G, Chen Y (2014) Decomposition of organic carbon in fine soil particles is likely more sensitive to warming than in coarse particles: an incubation study with temperate grassland and forest soils in Northern China. PloS One 9(4):e95348
2. Sarkar S, Leo BF, Carranza C, Chen S, Rivas-Santiago C, Porter AE, Schwander S (2015) Modulation of human macrophage responses to *Mycobacterium tuberculosis* by silver nanoparticles of different size and surface modification. PloS One 10(11):e0143077
3. Clapp RB, Hornberger GM (1978) Empirical equations for some soil hydraulic properties. Water Resour Res 14(4):601–604
4. Tyler SW, Wheatcraft SW (1992) Fractal scaling of soil particle-size distributions: Analysis and limitations. Soil Sci Soc Am J 56(2):362–369
5. Yang PL, Luo YP, Shi YC (1993) Soil fractal character token by particle-mass distribution. Sci Bull J 38:1896–1899
6. Wei-hua HGHZ (2002) Fractal property of soil particle size distribution and its application. Acta Pedologica Sinica 4
7. Filgueira RR, Fournier LL, Sarli GO, Aragon A, Rawls WJ (1999) Sensitivity of fractal parameters of soil aggregates to different management practices in a Phaeozem in central Argentina. Soil Tillage Res 52(3–4):217–222
8. Wei Y, Wu X, Cai C (2015) Splash erosion of clay–sand mixtures and its relationship with soil physical properties: the effects of particle size distribution on soil structure. CATENA 135:254–262
9. Wang D, Fu B, Zhao W, Hu H, Wang Y (2008) Multifractal characteristics of soil particle size distribution under different land-use types on the Loess Plateau, China. CATENA 72(1):29–36
10. Martínez-Casasnovas JA, Sánchez-Bosch I (2000) Impact assessment of changes in land use/conservation practices on soil erosion in the Penedès–Anoia vineyard region (NE Spain). Soil Tillage Res 57(1–2):101–106

11. Islam KR, Weil RR (2000) Land use effects on soil quality in a tropical forest ecosystem of Bangladesh. Agr Ecosyst Environ 79(1):9–16
12. Deng Y, Cai C, Xia D, Ding S, Chen J (2017) Fractal features of soil particle size distribution under different land-use patterns in the alluvial fans of collapsing gullies in the hilly granitic region of southern China. Plos One 12(3):e0173555
13. Lyu X, Yu J, Zhou M, Ma B, Wang G, Zhan C, Wang D (2015) Changes of soil particle size distribution in tidal flats in the Yellow River Delta. PLoS One 10(3):e0121368
14. Buchan GD, Grewal KS, Robson AB (1993) Improved models of particle-size distribution: An illustration of model comparison techniques. Soil Sci Soc Am J 57(4):901–908
15. Skaggs TH, Arya LM, Shouse PJ, Mohanty BP (2001) Estimating particle-size distribution from limited soil texture data. Soil Sci Soc Am J 65(4):1038–1044
16. Parra-González SD, Rodriguez-Valenzuela J (2017) Determination of the soil quality index by principal component analysis in cocoa agroforestry system in the Orinoco Region, Colombia. J Agricult Ecol Res Int 1–8

Predictive Models for Load Reduction in Wireless Sensor Networks

Arun Avinash Chauhan and Siba K. Udgata

Abstract In this paper, we propose techniques for load reduction in Wireless Sensor Networks (WSNs) that work by understanding the nature of data. These techniques analyse time series data to understand patterns in the data and create simple or complex prediction models. These models are adaptive in nature, and are present both at the sensor nodes and the sink node. These models predict data whenever the sensor nodes measure data. Data transmission from sensor nodes to sink node occurs only when the measurements do not agree with model predictions. This reduces the amount of data transmitted across the network, leading to reduced communication and energy consumption. The prediction models are developed before sensor node deployment, and once steady-state operation starts, the prediction models are recreated at the sink node whenever their performance degrades. The sink node then sends updated model parameters to the corresponding sensor node whose model performance has degraded. Simulation results indicate a reduction of up to 88% in data transmitted from sensor nodes across the network, proving the efficacy of predictive models in reducing the amount of data sent, thereby saving transmission energy and improving network lifetime.

Keywords Load reduction · Wireless sensor networks · Predictive modelling · Machine learning

1 Introduction

The usual procedure to load reduction in WSNs is data reduction by aggregation. Simply put, aggregation can be defined as the ability to summarize information [1]. Most of these approaches take advantage of data similarity to aggregate data

A. A. Chauhan · S. K. Udgata (✉)
School of Computer and Information Sciences, University of Hyderabad,
Hyderabad 500046, India
e-mail: udgata@uohyd.ac.in

A. A. Chauhan
e-mail: avinashch@uohyd.ac.in

when data from disparate sources is routed over the network through a common routing path. This is called in-network aggregation. Two data packets of similar nature arriving at a sensor node in the network can be aggregated as one outgoing packet. Consequently, these procedures apply methods to overlap the paths or routes traversed by data from sensor nodes to the sink node. Sensor nodes on the overlapping paths that route data to the sink aggregate data from various sensor nodes. On the other hand, our proposed predictive modelling-based approach works at the source of the data—the sensor nodes that sense and send data. This way, rather than creating data packets and aggregating it over the network, we lessen the number of data packets being created and decrease the network load at the source. This is done with the help of predictive models present at the individual sensor nodes and the sink node. These models take advantage of the temporal redundancy in sensor node data and inherent reproducible characteristics of the time series data being measured. To this purpose, we create prediction models which helps let us know if the data measured at an individual node is expected or unexpected. These models reside at the individual sensor nodes (one model for each sensor node) and the sink node (multiple models, each corresponding to one sensor node in the network) and will help reduce the data sent by only sending data when the data measured is unexpected. The sink node has a copy of the model corresponding to each sensor node and refers its copy of the model to recreate the expected data, thus reducing the amount of transmitted data. In our previous work [2], we showcased a sink-driven predictive modelling approach to leverage the energy and processing capabilities of the sink node. In this work, we extend the predictive models suggested in [2], and provide comparative results across multiple predictive models, validating our work on two real-world datasets. The paper is organized as follows: In Sect. 2, we give a background on load reduction and related work, Sect. 3 presents the methodology employed by us, Sect. 4 presents the results of our work, before Sect. 5 concludes this paper and provides pointers for future work.

2 Background

Prediction-based data aggregation is usually categorized into three types: Stochastic approaches, time series forecasting, and algorithmic approaches [3]. Stochastic approaches employ probabilistic and statistical properties of the sensed data to find a model. Time series-based approaches adopt time series models which, although simple, have good accuracy in many practical applications. Algorithmic approaches employ heuristics to exploit the behavioural characteristics of sensing data to predict aggregate data [4]. Our technique is a time series-based approach.

One of the first attempts to use time series-based prediction model to aggregate data was by [5]. They proposed using the ARIMA model for data aggregation of time series data. ARIMA models are present at sensor nodes and CHs and whenever the predictions do not match the measurements, updated model params are sent to the CHs. Reference [6] further used ARMA in conjunction with compressive sensing

based on discrete cosine transform to make predictive models, with the assumption that the data is piecewise linear. Additional work on static prediction models which are present at sensor nodes and are used to aggregate data is presented in [7]. Other times, several simple techniques are used for modelling [8]. Here, five different data prediction models—ARMA, AR, MA, nonlinear regression, and exponential smoothing are used. The prediction accuracy of the five models is compared and the best model is chosen. Recently, [9] created a static but powerful model based on 1-D CNN and Bi-LSTM which picked features based on measurement correlation and modelled the data. Among recent studies, [10] provide a comprehensive comparison of machine learning algorithms used for predictive modelling in temperature prediction. 36 machine learning algorithms are compared for accuracy. Apart from learning that extraTrees work best for (indoor) temperature prediction, we come to find validation to what we observed during literature review. That is, the horizon or forecast period is usually less, up to 3 h, and that the prediction models are not dynamic.

We proposed a localized load reduction technique which uses predictive models. However, our models are adaptive in nature, and model recreation happens at the sink node [2]. Sink node transmits the models to sensor nodes whenever the models are updated. The entire operation is sink node driven. In this manner, we harness the greater processing and battery capabilities of the sink node, and save processing and transmission energy of the individual sensor nodes. Additionally, our forecast horizon (one year) is much longer than any existing in the literature. Here we extend our work by showcasing multiple models, ranging from simple averaging-based models to complex TBATS model, with a quick comparative study. Results show that our models predict data with sufficient accuracy and a reduction of up to 88% in payload data transmission of sensor nodes.

3 Methodology

3.1 Sensor Environment

We consider sensor nodes to be either randomly scattered or placed in a predetermined fashion over an area of interest. The nature of deployment does not matter since our techniques work at the source sensor nodes and destination sink node. We work with the following assumption:

– The underlying network is robust in that it handles packet loss.

The major components involved in the operation are sensor nodes, the sink node, and the temporal models—that are housed in at each of the sensor nodes and sink node. The WSN is configured as a time-driven or periodic WSN. In a PWSN, sensor nodes transmit measured data once every period. This period can be broken down into a certain number of slots, where a measurement is made once every slot. Generally,
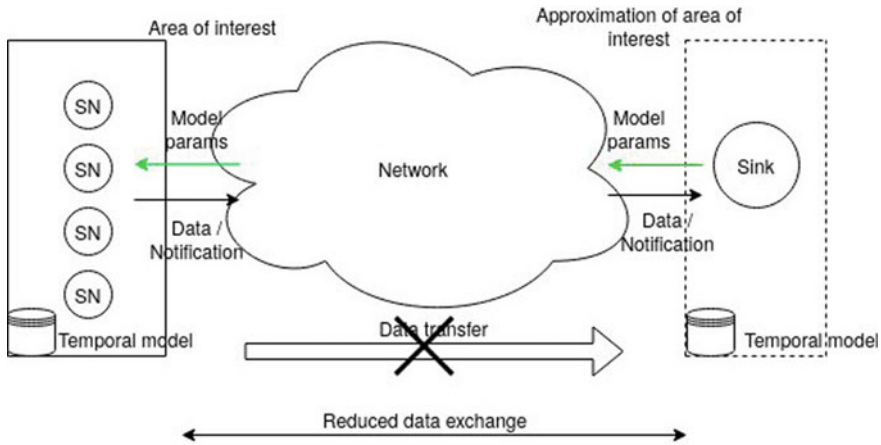
Fig. 1 System model

at the end of a period, a set of measurements is transmitted to the sink. Figure 1 showcases our approach. Predictive models are housed at the sensor nodes (left of image) and sink node (right of image). The whole idea or philosophy is to approximate the monitoring area at the sink node using predictive models. The approximated area is represented by the dotted rectangle. Without predictive models, there would have been major data transfer (shown as a solid arrow). Sink-driven predictive modelling instead leads to a much-reduced data usage (represented by a double-headed arrow). Unexpected data is sent from sensor nodes and updated model parameters are sent back from the sink node whenever the predictive model performance deteriorates at any sensor node.

The data measured by the sensor nodes comes from two real-world data sets, namely the five cities data set [11], and the SML2010 data set [12]. The five cities data set consist of outdoor meteorological data measurements from five Chinese cities over a period of six years (2010–2015). It consists of 16 features and 264,270 instances. The SML2010 data set consists of indoor meteorological data measurements recorded over 40 days at SML house at the Universidad CEU Cardenal Herrera (CEU-UCH). It consists of 24 features and 4137 instances.

3.2 Load Reduction

Load reduction and reconstruction Our technique consists of two interworking aspects. At one end, a predictive model at each sensor node predicts the observation once every time slot, while the sensor measures the observation in that time slot. When the period elapses, the sensor node sends the measurements to the sink node only if the measurements and predictions mismatch. At the other end, there is a

Fig. 2 Operation in a nutshell

sink node that contains predictive models corresponding to each sensor node in the network. Once every slot, these models make predictions of observations at each of the sensor nodes in the network. At the end of the time period, if a sink node does not receive any data packet from any sensor node, it understands that the predictive model for that sensor node did not encounter any mismatch and is working as expected. Occasionally, the predictive models do not predict accurately and the sensor nodes send data packets in those instances. Should any sensor node(s) send data packets for a predefined number of consecutive time periods, the sink node updates the model corresponding to that sensor node. The algorithms involved are mentioned in [2]. Figure 2 depicts the steps involved in a nutshell.

Predictive model The predictive models play a primary role in data reduction. It is a mathematical model that predicts or estimates the value of data at any particular time interval. The attribute whose value is to be predicted is called the *dependent* variable and the attributes used in the model as inputs are called the *independent* variables. A model makes a prediction of a new observation based on what it learns from hundreds, thousands and sometimes even millions of prior observations in time. This is called historical or training data that is used to *train* the model. The data is time series data. Time series data is data that is *usually* recorded or measured at regular intervals of time. The measured data is chronologically ordered. The predictive model is also called temporal model when it has time as one of its independent variables. Time series data is analysed to understand underlying patterns and structure in data. This analysis helps understand what external parameters does the data depend upon and the characteristics of data. The usual components of time series data are trend, seasonality and remainder. Once the data is analysed, we build a mathematical model that helps predict the data.

Data preprocessing and transformation Time series data is firstly preprocessed. This involves using correlation to understand which attributes need to be used in our models. Once we identify the attributes needed, we check for NA or missing

values in these attributes and replace them with the last occurring value. We then check for anomalies and the values using the Twitter Anomaly Detection toolkit [13] and replace the anomalies with last proper value. Once the data is preprocessed, we divide the data into training data and test data. To divide the data, we determine the frequency of the data, and use twice the frequency value as the number of records in the training set. When that is not sufficient to train the model, we pick half the amount of data as training data, and the other half as the test data. In some scenarios, we need to transform data before modelling. Data that is not stationary needs to be transformed to make it stationary. Stationary data is time series data whose properties do not change with time. Presence of trend and seasonality in time series data makes the data nonstationary. Stationarity of the data is tested using unit tests like the Kwiatkowski–Phillips–Schmidt–Shin (KPSS) test. The data is then transformed to make the data stationary. These transformations are necessary in models like ARIMA which do not use trend and seasonality. On the other hand, models based on exponential smoothing use trend and seasonality. The two popular transforms are log transform and differencing data.

Simple models, Holt-Winters and TBATS In [2], for predictive modelling, we proposed decomposition of time series data and applying M5P regression tree for the respective components. In this paper, we went ahead and compared the performance of M5P regression tree against a few other techniques, namely—a simple averages and weighted averages-based technique, then Holt-Winters and ETS, and finally TBATS.

Simple model: averages and weighted averages A simple functional model when we have history data available is averaging and weighted averaging. A prediction is made on the basis of averaging of the values of previous observations made at the same point in time, every season, or period in the past. For instance, in five cities dataset, which has annual seasonality, the prediction of temperature on current day (date) is the average of previous years' measurements of temperature taken on the same date. To accommodate long-term changes, we can also choose weighted averaging where we give higher weight to temperature measurements/observations from the most recent observation made at the same time/date in all previous periods/seasons in history data. The weights decrease as the observations are further away from most recent observation. In case the data has multiple seasonalities, for instance, the five cities dataset has both a daily and an annual seasonality, we give higher weight to the observation corresponding to the smaller seasonality (day) and lesser weight to the observation corresponding to the larger seasonality (annual). In other words, the temperature at the same time yesterday will have a greater weight than the temperature measurement from the same time a year back. Results of this approach are surprisingly good considering the minimalist nature of the model. Based on the nature of the data being modelled, this model can be used in data forecasting when one needs an approximate prediction and the underlying WSN application does not have hard quality and accuracy bounds.

Holt-Winters and ETS For one-step forecasting, ETS performs better than some advanced machine learning methods like Long Short Term Memory (LSTM) and

Recurrent Neural Networks (RNNs) [14]. It is this finding that makes us explore and use simpler forecasting techniques like ETS and Holt-Winters in an iterative manner to enable long-term forecasting over a yearly prediction horizon. Holt-Winters is also known as Triple Exponential Smoothing because it uses exponential smoothing on the three components of the time series, i.e. seasonality, trend, and remainder. Determining the right frequency/seasonality of the data is crucial for Holt-Winters or ETS-based forecasting. Holt-Winters, although a simple technique, is surprisingly powerful. Before applying Holt-Winters or ETS, we first check if the training data needs to be differenced first. Once the data is differenced, it is converted to a time series object. The time series object is then passed as a parameter to the Holt-Winters function hw() or the ETS function ets() to create the model. Once the model is created, steady-state operation commences. The model is referred to iteratively, at every time slot, to make a one-step prediction. If the training data was differenced, every prediction is readjusted to account for the initial differencing.

TBATS An advanced forecasting model called TBATS developed by [15] uses combination of Fourier terms with an exponential smoothing state space model and a Box–Cox transformation. TBATS is usually applied for time series with complex seasonality. Complex seasonality is when the time series data has multiple overlapping seasonalities or high-frequency data. The complex seasonality in the data is succinctly captured using Fourier terms. This usage of Fourier terms also helps TBATS overcome the shortcomings of Holt-Winters and ETS which cannot handle high seasonality values. SML2010 dataset, for instance, has a high seasonality/frequency of 96 and cannot be accommodated by Holt Winters and ETS.

Sliding training data window Iterative or one-step modelling-based approaches use a sliding training data window. If the data packets are sent consecutively for a threshold number of times, the model is retrained using the sliding window at the sink. This sliding window initially would have consisted of the actual training data. Over time the sliding window slides right to accommodate incoming measurements at the sink node or predictions made by the model at the sink node. The oldest training data values drop off the left side of the sliding window. The sliding training window enables for adaptive modelling and long-term forecasts of up to a year and beyond, if need be.

4 Results and Discussion

4.1 Implementation

We tested our hypothesis using RStudio [16] which is an integrated development environment for R. The creation of the temporal models, the simulation WSN involving transfer of reduced data across the network, and the update of the models from sink node to sensor nodes were all coded with the help of R script files in RStudio. The

data in the simulation is transmitted using IEEE 802.15.4 packets [17]. There are four types of frames defined but we concern ourselves only with the data frame. The data frame consists of a header of 24 bytes and a payload section which can be of variable length to a maximum of 104 bytes.

4.2 Results and Discussion

We evaluated our predictive models using *R*-squared value, Root Mean Square Error (RMSE) and Mean Average Error (MAE). RMSE is calculated using the following Eq. 1.

$$\text{RMSE} = \sqrt{\frac{1}{n}\Sigma_{i=1}^{n}\left(\frac{d_i - f_i}{\sigma_i}\right)^2} \tag{1}$$

One can observe that the equation is similar to Euclidean distance equation—and can be thought of as the distance between vector of predicted values and vector of observations. In terms of a benchmark to compare the performance of our models, [18] report a *1-day out* RMSE of approximately 2.8 °C. *R*-squared, on the other hand, tells us how much the linear predictive model explains the response variable variation, in terms of percentage. Values of *R*-squared lie between 0 and 100%, or 0.00–1.00. 0.00 indicates that the model does not explain any of the variability in the predictions, whereas 1.00 indicates the model explains all variability. A good model has a value closer to 1.00. Mean Absolute Error (MAE) is the mean of the absolute error, where the absolute error is the difference between the measured value and true value. MAE is calculated using the following Eq. 2

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

Tables 1 and 2 tabulate the performance of the three models compared against the M5P regression tree-based model. It can be seen that when the horizon increases, the performance of the M5P regression tree-based model goes down, especially in the case of five cities dataset which has higher data variance (63.26) when compared to the SML2010 dataset (8.14). Iterative modelling approaches which use a sliding training data perform best in both datasets. Surprisingly, the simple model performs reasonably well and when the data to be modelled does not vary much, the simple model can be used as a predictive model. Note that Holt-Winters and ETS cannot be applied to SML2010 dataset because it has a data frequency of 96, and Holt-Winters and ETS cannot handle high-frequency data.

The saving in the amount of data transmitted depends on the difference threshold set to determine a mismatch. In the case of SML2010 dataset, for a difference threshold of 1.5, we had a saving of 75.97%, for a difference threshold of 2, the savings

Table 1 Performance of some of the techniques on five cities dataset

Technique	Horizon	Correlation	RMSE	R-squared	MAE
M5P	24	0.9621909	4.3796231	0.9258113	3.7691047
M5P	8760	0.0655896	11.3418243	0.004302	9.1736524
Simple model	8760	0.8947466	3.7211379	0.8005715	2.8806317
Applying TBATS	8760	0.8977154	6.5705631	0.8058929	5.6817308
Holt-Winters (Iterative)	8760	0.9608393	2.2361231	0.923122	1.6254993
ETS (Iterative)	8760	0.9607005	2.2365952	0.9229454	1.6291573
TBATS (Iterative)	8760	0.960025	2.259002	0.921648	1.647230

Table 2 Performance of some of the techniques on SML2010 dataset

Technique	Horizon	Correlation	RMSE	R-squared	MAE
M5P	96	0.9671215	1.8547682	0.9353241	1.5892308
M5P	1344	0.544716	2.2380737	0.2967155	1.7636186
Simple model	1344	0.9717119	1.4921725	0.9442241	1.2950554
Applying TBATS	1344	0.7757708	3.0613081	0.6018203	2.6959502
Holt-Winters (Iterative)	1344	NA	NA	NA	NA
ETS (Iterative)	1344	NA	NA	NA	NA
TBATS (Iterative)	1344	0.8600512	1.4595792	0.7396881	1.0916691

went up to 83.33% and for a difference threshold of 2.5, the saving in the amount of data transmitted between sensor nodes and sink node was 87.43%. In the case of five cities dataset, for a difference threshold of 1.5, we had a saving of 60%, for a threshold of 2, a saving of 69.1%, and for a threshold value of 2.5, we had a saving of 75.83% in the amount of data transmitted from sensor nodes to the sink node.

5 Conclusions

In this paper, we built upon the idea of using adaptive time series prediction models to predict data and reduce transmissions proposed by us in [2]. We implemented a sliding training data window to iterate one-step forecast and achieve adaptive, accurate long-term forecasts. All of this is sink driven, thus leveraging greater energy and processing capabilities at the sink node. We explored three more types of predictive models to go along with M5P regression tree model in our earlier work. We were able to achieve long-term forecasting, in some cases up to an year, and consequent load reduction. Over this period, we were able to reduce the payload data transmissions from the sensor nodes to the sink node by up to 88%. This translates to reduced transmission energy and reduced energy utilization at individual sensor nodes.

References

1. van Renesse R (2003) The importance of aggregation. In: Schiper A, Shvartsman AA, Weatherspoon H, Zhao BY (eds) Future directions in distributed computing. Lecture notes in computer science, vol 2584. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-37795-6_16
2. Chauhan AA, Udgata SK (2021) Load reduction using temporal modeling and prediction in periodic sensor networks. In: Udgata SK, Sethi S, Srirama SN (eds) Intelligent systems. Lecture notes in networks and systems, vol 185. Springer, Singapore. https://doi.org/10.1007/978-981-33-6081-5_20
3. Anastasi G, Conti M et al (2009) Energy conservation in wireless sensor networks: a survey. Adhoc Netw 7(3):537–568. https://doi.org/10.1016/j.adhoc.2008.06.003
4. Zhang H, Zhang X, Sung DK (2015) Lightweight self-adapting linear prediction algorithms for wireless sensor networks. IEEE Sens J 15(5):3050–3058. https://doi.org/10.1109/JSEN.2014.2385854
5. Li G, Wang Y (2013) Automatic ARIMA modeling-based data aggregation scheme in wireless sensor networks. J Wireless Com Netw 2013:85. https://doi.org/10.1186/1687-1499-2013-85
6. Thapliyal A, Kumar R (2016) Temporal compression in wireless sensor networks using compressive sensing and ARMA modeling. In: 2016 2nd international conference on advances in computing, communication, and automation (ICACCA) (Fall), pp 1–4. https://doi.org/10.1109/ICACCAF.2016.7748967
7. Shaban KB, Kadri A, Rezk E (2016) Urban air pollution monitoring system with forecasting models. IEEE Sens J 16(8):2598–2606. https://doi.org/10.1109/JSEN.2016.2514378
8. Yang M (2017) Data aggregation algorithm for wireless sensor network based on time prediction. In: 2017 IEEE 3rd information technology and mechatronics engineering conference (ITOEC), pp 863–867. https://doi.org/10.1109/ITOEC.2017.8122476
9. Cheng H, Xie Z, Shi Y, Xiong N (2019) Multi-step data prediction in wireless sensor networks based on one-dimensional CNN and bidirectional LSTM. IEEE Access 7:117883–117896. https://doi.org/10.1109/ACCESS.2019.2937098
10. Alawadi S, Mera D, Fernández-Delgado M et al (2020) A comparison of machine learning algorithms for forecasting indoor temperature in smart buildings. Energy Syst. https://doi.org/10.1007/s12667-020-00376-x
11. Liang X, Li S, Zhang S, Huang H, Chen SX (2016) PM2.5 data reliability, consistency, and air quality assessment in five Chinese cities. J Geophys Res Atmos 121. https://doi.org/10.1002/2016JD024877
12. Zamora-Martínez F, Romeu P, Botella-Rocamora P, Pardo J (2014) On-line learning of indoor temperature forecasting models towards energy efficiency. Energy Build 83:162–172. https://doi.org/10.1016/j.enbuild.2014.04.034
13. Twitter/anomaly detection with R. https://github.com/twitter/AnomalyDetection. Accessed 29 Oct 2021
14. Makridakis S, Spiliotis E, Assimakopoulos V (2018) Statistical and machine learning forecasting methods: concerns and ways forward. PLoS ONE 13(3):1–26. https://doi.org/10.1371/journal.pone.0194889
15. De Livera AM, Hyndman RJ, Snyder RD (2011) Forecasting time series with complex seasonal patterns using exponential smoothing. J Am Stat Assoc 106(496):1513–1527. https://doi.org/10.1198/jasa.2011.tm09771
16. RStudio Team. RStudio: integrated development for R. http://www.rstudio.com/. Accessed 29 Oct 2021
17. Kabara J, Calle M (2012) MAC protocols used by wireless sensor networks and a general method of performance evaluation. Int J Distrib Sens Netw 8(1):1–11. https://doi.org/10.1155/2012/834784
18. Holmstrom M, Liu D (2016) Machine learning applied to weather forecasting

Smart Wheelchair Using Brain Waves Through Machine Learning

Jenamani Chandrakanta Badajena, Srinivas Sethi, Amrit Dash, Priyanka Rout, and Ramesh K. Sahoo

Abstract Maneuvering a mechanical wheelchair is a difficult task for a paralyzed person. Hence, there is a need for designing a wheelchair that is intelligent and provides easy maneuverability for persons who are not capable of handling the manual maneuvering process. Our proposed system is designed to receive, process, and classify electroencephalographic signals before controlling the wheelchair. This paper is based on an analysis of the cognitive function of the human brain, and its deployment through machine learning algorithms. It has been analyzed that machine learning algorithms improve the accuracy of electroencephalograph (EEG) response data. We have captured brain signals using the NeuroMAX-32 instrument from human beings under various stimuli conditions and tried to classify data using naive Bayes, support vector machine (SVM), and decision tree (J48). Attention and meditation level of person has been obtained from EEG response data, and it will be used to move, control, and stop the wheelchair using microcontroller.

Keywords Machine learning · EEG sensor · Brain wave · Smart wheelchair · Naïve Bayes · Support vector machine · Decision tree

1 Introduction

Attention can be considered as a focal point to capture the brain waves of a person that can further be used to process and analyze signals for utilization in the implementation of autonomous smart wheelchairs [1–3], where the human makes decisions and the smart control technology helps in the automation of motion. The primary contribution of this research is fourfold: The first fold is to observe the brain activity. The second is to create a firmly established environment with different situations. The third is to be evaluated through a machine learning algorithm, and the fourth is to deploy the idea in

J. C. Badajena (✉)
Utkal University, Bhubaneswar, Odisha, India
e-mail: chand.cet@gmail.com

J. C. Badajena · S. Sethi · A. Dash · P. Rout · R. K. Sahoo
Indira Gandhi Institute of Technology, Sarang, India

the smart wheelchair to perform the task. Brain waves control an autonomous smart wheelchair, in which the human makes decisions to move the wheelchair (forward, backward, right, left, and stop) with smart controls and automates the motion. The advancements in a smart wheelchair are equipped with environmental sensors, HD camera, 3D scanner, Bluetooth device, and a computer-based system as the main processing unit and place a strong emphasis on computer cluster architecture. Besides their disability, intelligence technique is added to the wheelchair platform around user control.

The goal of our study is to develop a BCI framework that may be extended to offer more natural and involving control. This involves the creation, testing, evaluation, and deployment of a new data analysis method for the study of human cognitive function along with the psychological and behavioral effects of a BCI system on the cognitive capacity of a human user. In this section, we have discussed the architecture of the model and working process of the model and the method of collection of data and analysis and implementation in machine learning and how to convert it into the command of a motor in the wheelchair. Furthermore, the accuracy of various data methods has been examined.

2   Background

Soleymani [4] proposed synchronized stimuli videos about unusual degrees of freedom. The author discovered two concepts for continuous emotion [5]: "long short-term memory recurrent neural network" (LSTM-RNN) and CCRF. They studied contamination and the effects of facial muscles using EEG signals. The authors have taken facial expressions to carry out the complementary observation of that EEG signal. The previous frame was used to track the supervised descent method while using the landmark (SDM). A conditional random field (CRF) had been used to build a probability-based model to classify sequential data and segment support. The concentration of the signal in the EEG [6] is used as the control signal for the wheelchair, and the signal is sent to the STM32 [7] to allow forward movement of the wheelchair. The emotion reflection in real time was collected and interpreted using a brain–computer interface (BCI) [8]. Negative and positive emotions were used to classify EEG data using k-NN [9] algorithm and multilayer perception neural network (MLPNN) [9]. The classifier algorithm sets up the channel selection. Individual participants used the EEG channel to perform various classifications. The resulting feature vector is being used to classify related negative and positive emotions using a related classifier.

k-NN classifier along with a new object was selected for feature extraction and distance calculation using Euclidean distance, Minkowski distance, and Hamming code. In all of the preceding research, the authors analyzed their observations in the context of an advertisement, a short film, or a non-biased situation.

Table 1 Comparison of various types of classified signals [6, 9]

Signal	Frequency (range) (Hz)	Amplitude (μV)	Originate lobe
Alpha	8–13	30–50	P-lobe/O-lobe
Beta	14–30	5–20	F-lobe/P-lobe
Theta	4–7	Less than 30	P-lobe/T-lobe
Delta	0.5–3	100–200	F-lobe/T-lobe
Gamma	31–50	5–10	F-lobe/T-lobe

Table 1 contains the various types of EEG signals with their respective frequency range and amplitude. These signals have been used further in developing and simulating the proposed BCI framework to capture perceptual and sensitive information from the temporal lobe, frontal lobe, parietal lobe, and occipital lobe [9].

3 Methodology

An experimental protocol was designed to capture participants' attention and meditation level which is shown in Figs. 1 and 2. Wheelchair will be moved, controlled, and stopped as per attention and mediation level of person. Attention and mediation level of person will be obtained from EEG response data received from brain using the NeuroMAX-32 instrument. The duration of the experiment is 900 s including preparation and rest. The experiment is based on visual movements tests and facial expression tests. These two factors were chosen because they have a great influence on participants' selective attention and meditation-related area. The experiment was conducted to study the paired habituation of children and adults and when the external force or anybody pinched the body that reflects in the attention and some change in facial expression which reflects the motion of a person getting hike. When the external force was applied to the human body, it was raised till the human body gets calm. When the human body gets relaxed and the mind is in a stable state, at
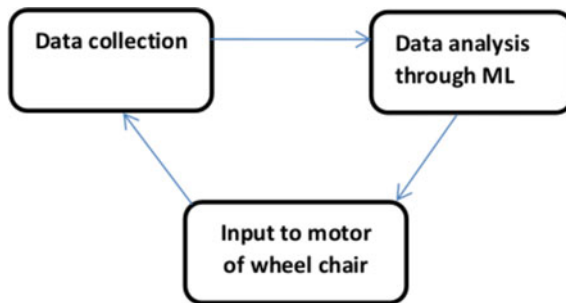
Fig. 1 Architecture of model
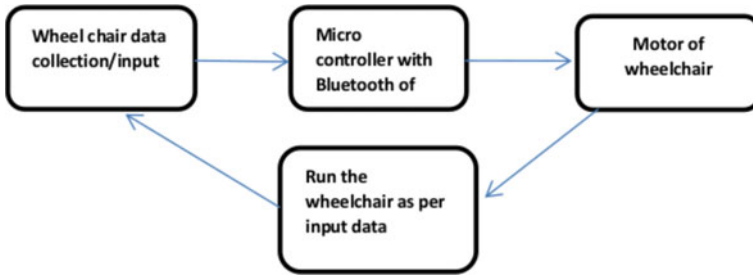
Fig. 2 Working process

that time the meditation of the human body gets hiked, and like this, the other lobes get hiked, and after some moment, it is normalized.

3.1 Architecture for Proposed Model

As per Fig. 1, the data was collected by the EEG machine (NeuroMax-32) from hundred participants and analyzed through machine learning algorithms, viz. SVM, decision tree (J48), and naive Bayes classifier. It, in turn, gives the command to the motor of the wheelchair, and according to command data, the wheelchair starts to work and the process will continue symmetrically.

3.2 Functionality of Smart Wheelchair

As per Fig. 2, the data has been collected from the human brain using EEG machine and analyzed and trained through machine learning teachniques. The trained data send to the microcontroller through Bluetooth of a wheelchair. Then the Microcontroller gives the command to the motor of the wheelchair and the wheelchair started working as per the data given by the Bluetooth.

3.3 Data Collection Framework

In this part, the proposed paradigm for categorizing emotions in various situations is described.

First, data from EEG signals was obtained from several lobe points of the brain using the NeuroMax-32 device. After that, the data was preprocessed in anticipation of extraction in the future. Then, by selecting the proper EEG channel, it generated a feature vector. The data was then examined and grouped into positive and negative.

Data collected from various lobes will be paired and analyzed. The right sides lobes, FP2-F8, F8-T4, T4-T6, T6-O2, and left sides lobes, FP1-F7, F7-T3, T3-T5, P3-O1 can be paired and analysed for different applications. In this work, two pairs such as, P3-O1 and T4-T6 provide EEG response data that reflects meditation and attention level of person, and it is in the range of 0–100.

Digital wavelet transform (DWT) is used to convert the extracted data and EEG signal. The model's last job involves categorization and prediction. It has been used during the analysis phase. Then, accuracy is obtained using the classifier.

3.4 Experimental Setup

The working mechanism of the approach has been expressed in three stages as per Fig. 3. In the first stage, the NeuroMax-32 channel headset has been used to capture the brain wave by using 32 electrodes placed in the human skull. It has 32 channel represented by 32 red and black connecting wires. The connecting wires have two ends; one end has been connected to the NeuroMax board, and another end has been connected to the electrodes. Out of 32, 16 electrodes are placed in right side of brain represented by red color wire and another 16 electrode are placed in left side of the brain represented by black color wire.

In the second stage, EEG signals will be received from brain by EEG headset through electrodes placed on the head and it will be sent to the NeuroMax board which in turn sent the captured signals to the connected computer through USB serial cable. The EEG signals are captured by NeuroMax software running in the computer, and output is represented in the graphical view.

Fig. 3 Experimental setup for the proposed model

In the third stage, the data in the graphical view is converted into an excel format that has been used for analysis in the proposed work.

3.5 Data Analysis and Modeling

The final step to control the wheelchair on the decision of the appropriate movement of a person is classification; this can be done with the implementation of machine learning algorithm, which is an application of AI. Machine learning models and applications can be categorized into supervised learning, unsupervised learning, and reinforcement learning.

In supervised learning, input data (X) and output data (Y) are a function of $Y = f(X)$. This has been resolved through the use of classification and regression. Regression problems occur when output data is used instead of a real or continuous value. Classification problems occur when data is output on a category basis. Decision tables, like decision tree classification models, are primarily used for machine learning algorithms.

Where W_1 and W_2 are the weight factor given to attention label A_L and meditation label M_L, respectively. We have given 70% focus to the attention level, and that is why we consider $W_1 = 0.7$, and whereas 30% focus is given to meditation level, and that is why we consider $W_2 = 0.3$ collectively to define the status of the person using attention and meditation using Eqs. 1 and 2.

A_L data obtained from T4-T6 lobe points (set of lobes considered for attention).
M_L data obtained from P3-O1 lobe points (set of lobes considered for meditation)

$$\text{Att\_med} = A_L * W_1 + M_L * W_2 \tag{1}$$

$$\text{Att\_med}_{\text{status}} = \begin{cases} \text{low} & \text{for Att\_med} < 35 \\ \text{medium} & \text{for } 35 < \text{Att\_med} < 55 \\ \text{high} & \text{for Att\_med} > 55 \end{cases} \tag{2}$$

In Eq. 1, Att_med represents mental status of person based on his attention and mediation level, and further, it has been categorized into low, medium, and high as reflected in Eq. 2. As per this, mental status of person speed of wheelchair will be controlled. Further, it can also control, stop, and move wheelchair. In this work, the wheelchair will stop when Att_med$_{\text{status}}$ is low, it will move slowly in medium status, and it will move in a steady way for high status. For this, we have studied the data using various machine learning approaches to judge the accuracy of model.

Processes to be followed

P1: Calculate the prior probability based on the given class labels.

P2: Find the likelihood probability for each class individually.

P3: Calculate the posterior probability using the naive Bayes formula.

P4: Find the class with the highest probability and label it as the probability class.

3.6 Machine Learning Algorithm Used for Classification

Machine learning can be helpful in the prediction of outcomes based on the inputs collected from various previous data without the need for explicit programming. It comprises sets of algorithms that can take data as input and can automatically improve user experience.

3.6.1 Naive Bayes Algorithm

Naive Bayes is a collection of "probabilistic classifiers" based on Bayes' theorem with a high amount of statistical data independence between features or elements taken for classification.

3.6.2 SVM

A support vector machine (SVM) is a learning algorithm based on a supervised learning method that can be helpful for data analysis, pattern recognition, and regression analysis. It is a classifier for the separation of the hyperplane. SVM is a non-probabilistic binary linear classifier. It is a supervised learning method associated with other learning algorithms. It can also be used as a discriminative classifier defined by separating hyperplane. It plays a significant role in the categorization between text and hypertext by accepting two possible classes from the input.

3.6.3 Decision Tree (J48)

A decision tree is another popular machine learning tool that uses a non-backtracking or greedy approach. The decision trees are constructed in a top-down or bottom-up manner. Initially, training is conducted with a set of tuples along with their associated class labels which are further recursively partitioned into smaller subsets.

3.7   *Performance Evaluation Parameter*

True positive (TP) rate, false positive (FP) rate, recall, precision, F-measure, mean absolute error (MAE), root mean squared error (RMSE), relative absolute error (RAE), and root relative squared error (RRSE) are the various parameters used for evaluation of machine learning algorithms used for classification in the proposed work.

For any potential parameter value chosen to differentiate between two classes or different cases, certain data will be correctly classified as positive (TP = true positive fraction) and some may be erroneously classified as negative (FN = false negative fraction).

The FP rate is calculated as FP/FP + TN, where FP stands for false positives and TN for true negatives (the total number of negatives is FP + TN). It is the chance that a false alarm will be triggered; a positive result will be returned when the true value is negative.

$$\text{Recall} = \frac{\text{TPR}}{\text{FN} + \text{TPR}} \tag{3}$$

$$\text{Precision} = \frac{\text{TPR}}{\text{FPR} + \text{TPR}} \tag{4}$$

$$F\,\text{Measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5}$$

Precision, recall, and F-measure have been determined as per Eqs. (3), (4), and (5). In statistical analysis of binary classification, the F-score or F-measure is a measure of a test's accuracy. An F-score can have a maximum value of 1.0, signifying perfect precision and recall, and a minimum value of zero if the precision or recall is both zero. Correctly classified instances are used for the compilation of true positive and true negative occurrences (TP + TN). For both false negative and false positive classifications (FP + FN), the same incorrectly classified instances were used.

The graph for performance evaluation that can be helpful in the classification of performance from the application of multiple classifiers is termed receiver operating characteristic (ROC). This is helpful in the demonstration of the relationship between the rates of true positives and false positives. ROC is a machine learning prediction concept that represents a diagonal approach. There are various parameters associated with the performance evaluation namely, mean absolute error (MAE), root mean square error (RSME), and relative absolute error (RAE). The MAE quantifies the average measurement of erroneous items in a given set of values where values are the distribution of probabilistic data items. The standard deviation or the root mean square error (RSME) provided information about the deviation of individual data items from the mean or in others words it gives a mathematical idea about how close the data points are toward the mean. Relative absolute error (RAE) is a method

for calculating the accuracy of an analytical model. RAE is an abbreviation for the absolute rate.

4 Result and Discussion

In *NeuroMax-32*, the lobe points *T4, T6, P3, and O1* are primarily taken as emotion and sentimental analysis. The data has been analyzed and computed using these lobe points. The data was gathered from various nerves of the brain from data pair points *T4-T6* and *P3-O1*. As a result, the data amalgamation technique can be a suitable mechanism for more accurate analysis. The data amalgamation was performed with the *T4-T6* pair of points and the *P3-O1* pair of points. Multiple paired lobes points are calculated from the 32 port NMX32 device. *P3, O1, T4,* and *T6* are primarily designed to assess emotion and sentiment. The data has been analyzed using these four lobes as a starting point. The data aggregation has been computed from these two pairs, as shown in Figs. 4 and 5. In these graphs, at the time of pinching on the skin of the person or heating to the skin of a person, the emotion may be observed through the segments of entire result.

As per Fig. 4, it has been observed that the resultant data is unstable due to outside activities on the human body, till the human body gets calm. From 185 to 227 of X-axis showing more value as a person gets pinched which reflects the emotion of a person getting hiked. Like this, the other points get hiked and normalized, and so on. The process will continue till the human body gets some external forces.

As per Fig. 5, it has been observed that the data becomes hike because the external force was applied on the human body and then it raised till the human body gets calm. Like in 1981–2080, the human body gets relaxed with the concentrated assignment. So, the meditation of the human body gets hiked. Like this, the other points get hiked and normalized, and so on.
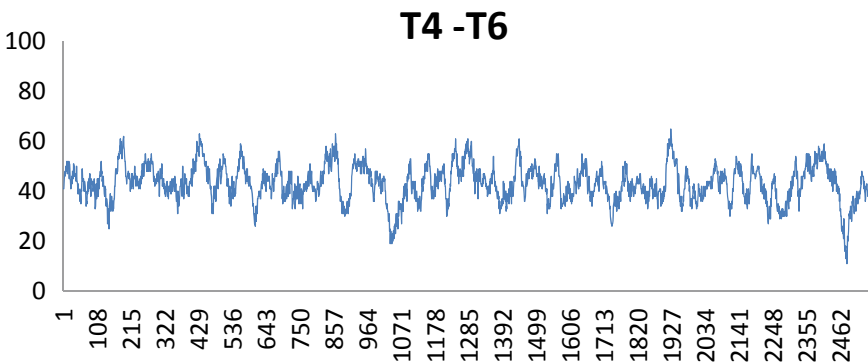
Fig. 4 Data aggression between T4 and T6
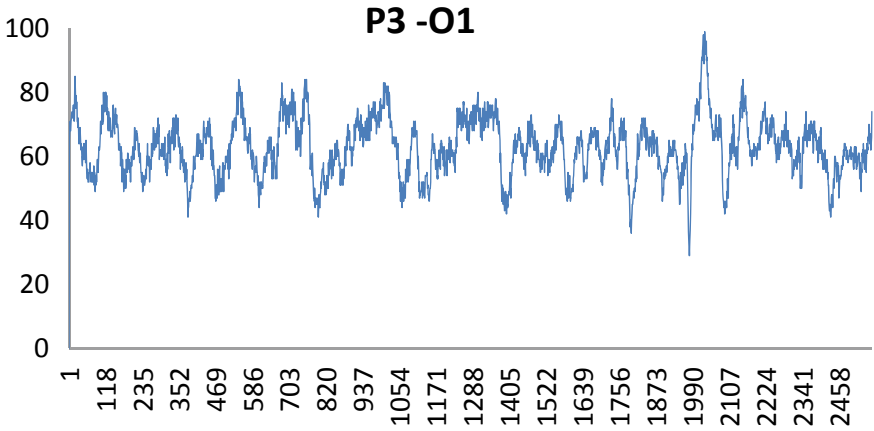
Fig. 5 Data aggression between P3 and O1

In Fig. 6, it is shown that the range of data in all methods is mostly medium-range data which was collected when the input body in a normal position, when the input body gets hurt by some external body or pinched by someone or surprised or scared the data level in all methods became high, and when the input body gets calm or sleepy the mind was in peace the data level becomes low.

In Fig. 7, we observed the kappa statistic, mean absolute error, and root mean squared error for naïve Bayes, SVM, and decision tree (J48) and we find that naïve
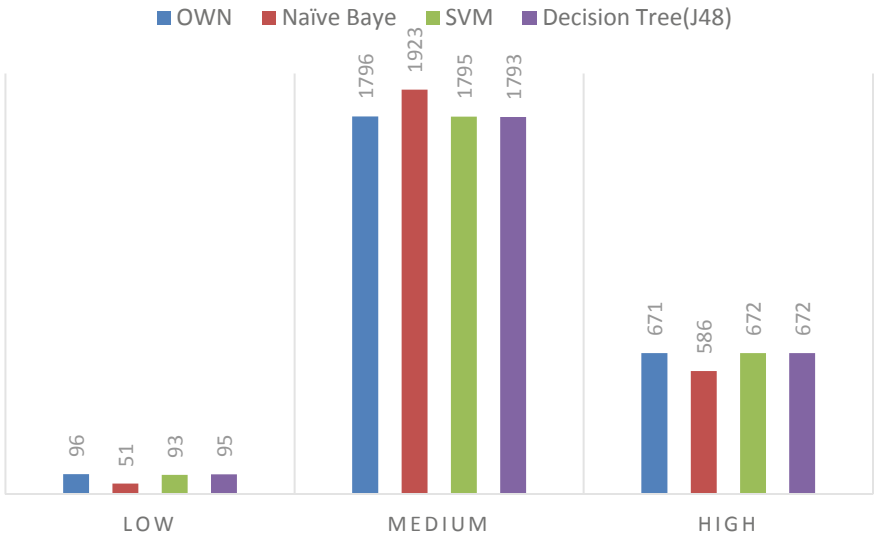
Fig. 6 Based on category
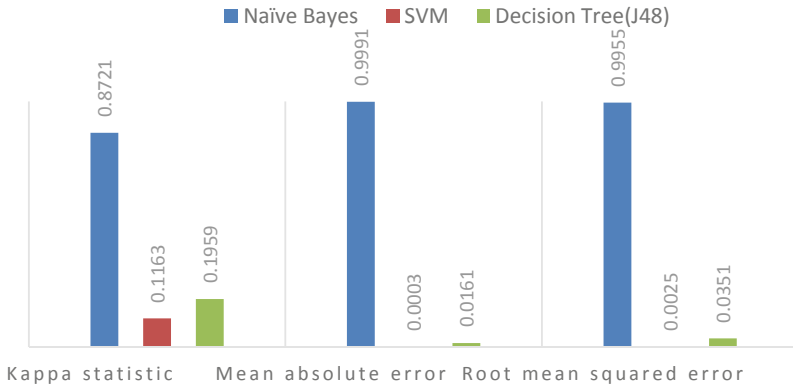
Fig. 7 Error detection

Bayes is more kappa statistic value than SVM and decision tree. And, also we find that SVM has very little error in comparison with naïve Bayes and decision tree.

In Fig. 8, we have analyzed the TP rate, FP rate, precision, recall, F-measure, MCC, ROC, and PRC area and we find that in SVM all the classes are the same except the FP rate. In SVM, the FP rate is zero and the decision tree is almost the same in all parameters expect FP rate; the FP rate is zero here. And, in the naïve Bayes, all the parameters are less than one.

As per Fig. 9, we can say that the accuracy level of SVM is higher in comparison with naïve Bayes and decision tree. The accuracy level in SVM is 99.96%, whereas in decision tree it is 99.80% and in Naïve Bayes it is 94.72%. The incorrectly classified instance is closer to 0 in both SVM and decision tree and a little in naïve Bayes, and also, we find that in SVM we get very less error in comparison with the decision tree and naïve Bayes.
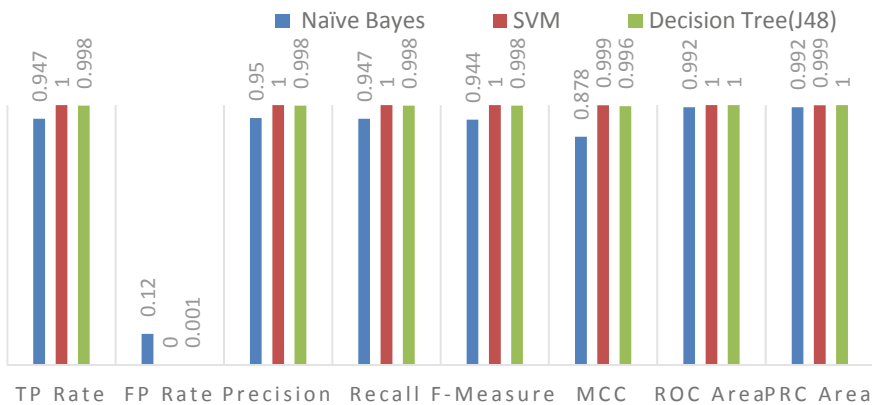
Fig. 8 Classifier analysis
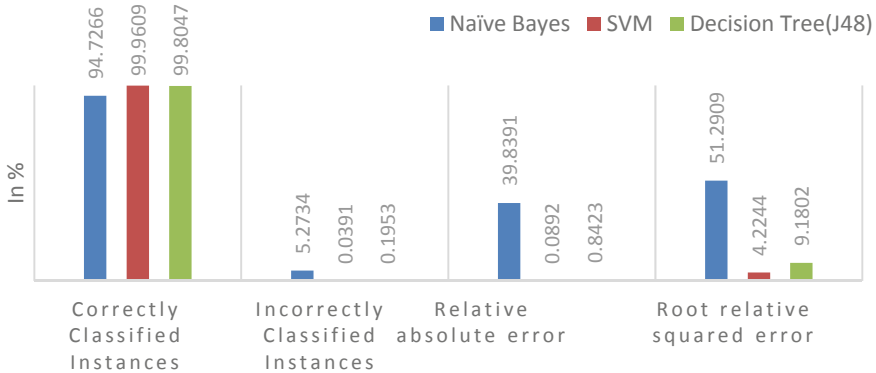
Fig. 9 Accuracy level

5 Conclusion

We have tried to develop a smart wheelchair that can work on behalf of an incapacitated human being by the methods discussed in this work. We have used various machine learning algorithms to design an artificial intelligence-based experience for people on wheelchairs. For the same, we have captured EEG response data from human brain using NeuroMax-32 headset attached with the skull to detect brain waves. Further, attention and meditation level has been measured using T4-T6 and P3-O1 lobe points of EEG response data. It has been used to determine mental status of person that will be used to move, control, and stop the wheel chair. In the future, we will try to move the wheelchair on left and right side using eye movement.

References

1. Xie Y, Li X (2015) A brain controlled wheelchair based on common spatial pattern. In: 2015 International symposium on bioelectronics and bioinformatics (ISBB), pp 19–22. https://doi.org/10.1109/ISBB.2015.7344913
2. Kim K-T, Suk H-I, Lee S-W (2018) Commanding a brain-controlled wheelchair using steady-state somatosensory evoked potentials. IEEE Trans Neural Syst Rehabil Eng 26(3):654–665. https://doi.org/10.1109/TNSRE.2016.2597854
3. Carlson T, Leeb R, Chavarriaga R, del Millán JR (2012) The birth of the brain-controlled wheelchair. In: 2012 IEEE/RSJ international conference on intelligent robots and systems, pp 5444–5445. https://doi.org/10.1109/IROS.2012.6386299
4. Soleymani M, Asghari-Esfeden S, Fu Y, Pantic M (2016) Analysis of EEG signals and facial expressions for continuous emotion detection. IEEE Trans Affect Comput 7(1):17–18
5. Sahu PK, Kumar Sahoo R, Sethi N, Sethi S (2020) Emotion classification based on EEG signals in a stable environment. In: 2020 International conference on computer science, engineering and applications (ICCSEA), pp 1–5. https://doi.org/10.1109/ICCSEA49143.2020.9132966

6. Khalifa WH, Salem A-BM, Revett K, Roushdy M (2012) A survey of EEG based user authentication schemes. In: 8th International conference on informatics and systems INFOS2012, May 2012
7. Xin L, Gao S, Tang J, Xu X (2018) Design of a brain controlled wheelchair. In: 2018 IEEE 4th international conference on control science and systems engineering (ICCSSE), pp 112–116. https://doi.org/10.1109/CCSSE.2018.8724794
8. Huang D, Qian K, Fei D-Y, Jia W, Chen X, Bai O (2012) Electroencephalography (EEG)-based brain-computer interface (BCI): a 2-D virtual wheelchair control based on event-related desynchronization/synchronization and state control. IEEE Trans Neural Syst Rehabil Eng 20(3):379–388. https://doi.org/10.1109/TNSRE.2012.2190299
9. Ozerdem MS, Polat H (2017) Emotion recognition based on EEG features in movie clips with channel selection. Brain Inform 4(4):241–252

Wi-Fi Signal-Based Through-Wall Sensing for Human Presence and Fall Detection Using ESP32 Module

Sahoo Ajit Kumar, K. Akhil, and Siba K. Udgata

Abstract Device-free Wi-Fi sensing has gained much attention due to its simplicity, low cost, and it requires no additional hardware. The main advantages of Wi-Fi sensing are, it is unobtrusive, can operate through walls, work without lighting, is ubiquitous, and does not require users to carry any additional wearable devices. These Wi-Fi signals can be exploited to detect the event and recognize various human activities. This paper proposes a device-free Wi-Fi sensing system to track human presence and fall activity detection across the wall using the Channel State Information (CSI) values extracted from the received Wi-Fi signals. The received signal characteristics changes with the presence of the human beings, and their activities influence the signal propagation, resulting from reflection and scattering. The activities can be recognized by analyzing the CSI values corresponding to different subcarriers of the received signal. CSI values contain fine grain information such as amplitude and phase to achieve better sensing accuracy with a unique pattern that can be observed corresponding to each activity. We developed the transmitter and receiver hardware modules together with the necessary software for capturing the CSI from Wi-Fi signals using low-power and low-cost ESP32 module. Multiple experiments are conducted for human presence and fall detection activities in indoor environments. The proposed model finds the best time window size to detect the activities instead of considering the usual fixed time window size. This helped to increase the detection accuracy. The results show the effectiveness of the proposed method with accurate event detection.

Keywords Wi-Fi sensing · Device free sensing · Channel state information (CSI) · Received signal strength (RSS) · Subcarriers · Activity detection · ESP32

S. Ajit Kumar · K. Akhil · S. K. Udgata (✉)
School of Computer and Information Sciences, University of Hyderabad,
Hyderabad 500046, India
e-mail: udgata@uohyd.ac.in

S. Ajit Kumar
e-mail: ajit@uohyd.ac.in

1 Introduction

Recent progress in Wi-Fi technology has established the use of Wi-Fi signals to sense the state of a person, such as sitting, standing, falling, walking, movement of body parts, and so on [1]. Wi-Fi signals often propagate through the direct path and multiple paths via reflection and scattering in indoor environments. The advantages of Wi-Fi sensing are: (1) it works with existing Wi-Fi infrastructures, as Wi-Fi devices are present almost everywhere and (2) noninvasive sensing can accomplish the sensing tasks without user awareness, and inconvenience [7]. The signal propagation paths can be broadly categorized into two folds: (1) Line-of-Sight (LoS) and (2) Non-Line-of-Sight (NLoS) as shown in Fig. 1. In the case of LoS, signals propagate through a direct path between the transmitter and the receiver. In the case of NLoS, the signal propagation may deteriorate as the signal travels through multipath reflections from transmitter to receiver [5].

Device-free Wi-Fi sensing is typically based on signals, namely RSS and CSI. RSS is MAC layer information that represents the strength of the received signal and is only a superimposition of multipath signals. It represents coarse-grain information that can be used for localization or to calculate the distance between the access point and receiver. RSS is susceptible to multipath effects and unable to resolve multipath propagation, which leads to unpredictable fluctuations, resulting low accuracy in sensing tasks [6]. To overcome the limitations of RSS based sensing, the CSI feature of Wi-Fi is adopted. Multiple-Input Multiple-Output (MIMO) and Orthogonal Frequency-Division Multiplexing (OFDM) provide channel state information, which represents the amplitude attenuation and phase shift of each subcarrier [12]. This CSI is the physical layer information of Wi-Fi and is more granular and stable than RSS. It can provide more fine grained information of the signal as it carries both amplitude and phase information across subcarriers [13]. CSI is widely used for wireless sensing purposes as it effectively manages the multipath effects at the subcarrier level. CSI contains the channel properties of a communication link such as channel frequency
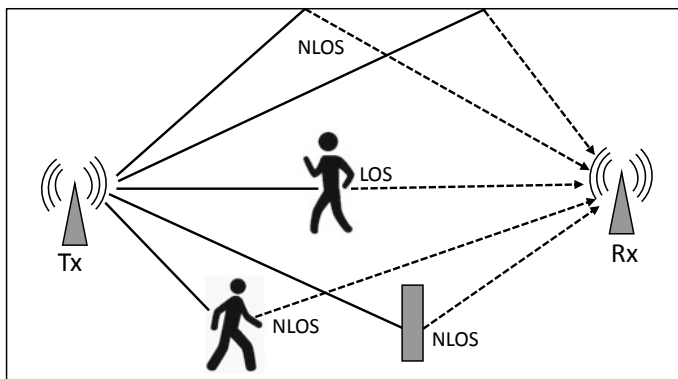
Fig. 1 A scenario of LoS and NLoS of Wi-Fi signal propagation

response of subcarriers and characterizes how the signal transmission between the transmitter and receiver [10]. CSI reveals the combined effect of scattering, fading, distance attenuation, environmental attenuation, time delay, amplitude attenuation, and phase shift of multiple paths on each communication subcarrier [16].

This work develops and demonstrates through-wall experiments of human presence and fall detection from CSI data gathered using the low-cost, low-powered ESP32 Wi-Fi module. We conducted experiments in the laboratory for human presence and fall detection with one wall between the transmitter and receiver Wi-Fi module. We have also conducted experiments for the same events within two walls by keeping the transmitter and receiver outside the walls. Repeated experiments are conducted to verify the effectiveness of the Wi-Fi sensing and to check the accuracy of event detection.

The remainder of this paper is organized as follows. Section 2 reviews the related works on CSI-based activity monitoring. Section 3 describes the theory of CSI and extraction of CSI data from the ESP32 Wi-Fi module. Experimental procedure, results, and analysis are presented in Sect. 4. Analysis of experimental CSI data to detect human presence and fall activity described in Sect. 5. Conclusions and future works are outlined in Sect. 6.

2   Related Works

This section describes the related works on CSI-based Wi-Fi sensing for human activity detection and monitoring. Ahmed et al. [1] presented a survey on machine learning approaches for device-free human gesture recognition using CSI metric of Wi-Fi devices. Atif et al. [2] presented ESP32 based Wi-ESP CSI tool for CSI measurements as well as the processing and its applications in device-free Wi-Fi sensing. Fang et al. [5] presented non-line-of-sight human presence detection from CSI measurements using commodity Wi-Fi devices. Hao et al. [7] proposed a complex human motion recognition method, namely, CSI-HC based on CSI and the average accuracy of the motion recognition in indoor environment up to 85.4%. Hernandez and Bulut [8] developed a smart phone based application to collect, process, and analyze the CSI data received from ESP32 device. Through-wall experiment of human presence as well as their moving directions using CSI data of ESP32 Wi-Fi device demonstrated by Hernandez and Bulut [9]. Liu et al. [11] presented a comprehensive review of Wi-Fi signals, theoretical models, signal preprocessing, segmentation, feature extraction, human activity monitoring and recognition, and applications. In depth survey on signal processing, algorithms, applications, and performance of Wi-Fi sensing as well as basic concepts, advantages, limitations are presented by Ma et al. [13]. Yang et al. [17] proposed a framework to improve human activity recognition based on different signal enhancement approaches. Human presence detection and event classification using support vector machine with 95.3% accuracy for event classification and 93.3% accuracy for human presence detection is presented by Li et al. [10]. Wang et al. [14] discussed a comprehensive overview of the working

principle and system architecture of device-free Wi-Fi sensing and its applications, challenges, and opportunities. Ma [12] presented a convolutional neural networks based approach for sign language recognition (SignFi) using CSI. Wang et al. [15] classify and analyze Wi-Fi CSI-based behavior recognition. A comprehensive survey of deep learning based wireless sensing systems for human detection is discussed by Zhang et al. [18].

3   Channel State Information

In this section, we present the basic concepts of channel state information. In Wi-Fi communication, channel state information represents the channel quality of a communication link. The OFDM of IEEE 802.11n standard provide the amplitude and phase difference information of different subcarriers in the form of CSI matrix. In the physical layer, OFDM divides the spectrum band into different frequency sub-bands called subcarriers [12, 17]. Since every subcarrier is independent of each other, the multipath effects are different for every subcarrier, which causes changes in the amplitude and phase. CSI also captures the various environmental changes. Therefore, any movement or changes in the environment can be detected by different subcarriers [13]. As a result, CSI can distinguish and identify different granularities of actions or movements accurately.

Multipath propagation also leads to constructive and destructive interference, frequency-selective fading, and shadowing. This multipath propagation can be depicted by Channel Frequency Response (CFR) in the frequency domain and Channel Impulse Response (CIR) in the time domain. CFR can be obtained by performing Fourier Transformation (FFT) on CIR. In the time domain, the CFR, $h(\tau)$ of Wi-Fi channel can be modeled as:

$$h(\tau) = \sum_{i=1}^{L} a_i e^{j\phi_i} \delta(\tau - \tau_i) \tag{1}$$

where a_i and ϕ_i represents the amplitude and phase shift of the ith path prospectively, τ_i is the time delay on ith path, L is the total number of multipaths, and $\delta(\tau)$ is the Dirichlet impulse function.

In the frequency domain, the channel response is modeled as CFR and can be stated as:

$$\mathbf{y} = \mathbf{Hx} + \mathbf{N} \tag{2}$$

where $\mathbf{x}$ and $\mathbf{y}$ are the transmitted and received signal vectors, respectively. $\mathbf{H}$ is the channel state information matrix. $\mathbf{N}$ is the additive white Gaussian noise vector.

Based on the drivers used in the Wi-Fi devices, the CSI will be split into different subcarriers. The matrix **H** can be expressed as Eq. 3.

$$\mathbf{H} = [\mathbf{H}_1, \mathbf{H}_2, \mathbf{H}_3, \dots, \mathbf{H}_n] \tag{3}$$

where n is the number of subcarriers, in this work, we use the ESP32 Wi-Fi module, and the number of subcarrier in CSI is 64. Here $\mathbf{H}_i$ is the vector containing complex number depicting the amplitude and phase captured for each subcarrier. $\mathbf{H}_i$ can be expressed as (4)

$$\mathbf{H}_i = |\mathbf{H}_i| e^{j \sin(\theta)} \tag{4}$$

where $|\mathbf{H}_i|$ and θ are the amplitude and phase response of the ith subcarrier, respectively.

3.1 ESP32 Wi-Fi Module for CSI Data Collection

This section introduced the key features of the ESP32 module used in this study as shown in Fig. 2 and its specifications. Espressif IoT development framework (esp-idf) recently added user-programmed software for ESP32 microcontroller to collect CSI data [3]. This ESP32 is a low-cost, low-power device with both Wi-Fi and Bluetooth functionalities [4]. The device can support 802.11 b/g/n protocols (802.11n (2.4 GHz) up to 150 Mbps) and can be configured to work either Station mode (STA) or Access Point mode (AP). In STA mode, the ESP32 acts as a Wi-Fi device that can connect to existing Wi-Fi networks, and in AP mode, the ESP32 acts as a host Wi-Fi network for other devices. The present implementation of ESP32 supports only one antenna for transmitter (Tx), and receiver (Rx) [8]. The ESP32 has two low-powered Xtensa 32-bit LX6 microprocessors, making it a powerful dual-core (core 0 and core 1) microcontroller. This device is a single 2.4 GHz Wi-Fi and Bluetooth combo chip designed for Internet of things (IoT) applications. It has 440 kB of ROM and 520 kB RAM and supports a clock frequency up to 240 MHz.

In ESP32, subcarriers CSI channel frequency response is computed when a packet is transmitted from Tx to Rx. The CFR of a subcarrier is of two bytes of character, the first segment is the imaginary, and the second segment is the real value [2]. ESP32 Wi-Fi supports bandwidths of HT20 and HT40, and the default bandwidth is HT40. HT20 supports a maximum bandwidth of 72 Mbps, and HT40 supports a bandwidth of 150 Mbps.
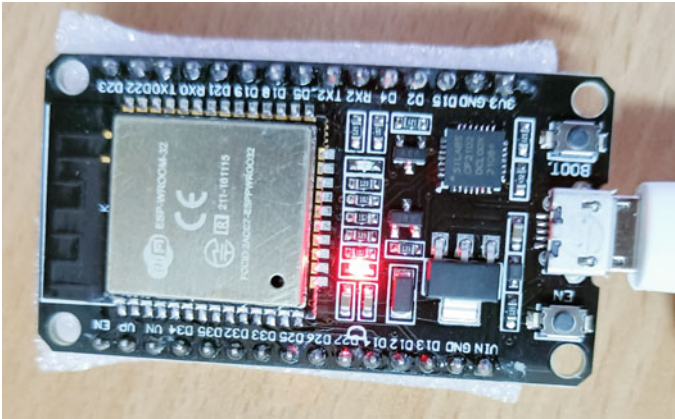
Fig. 2 ESP32 module used for CSI data collection

4   Experiments

We developed the hardware module using the ESP32 module for both transmitter and receiver using the IEEE 802.11n 2.4 GHz Wi-Fi communication standard. In this paper, we used low-cost and low-powered ESP32 Wi-Fi modules (cost is around 10 USD) for Wi-Fi sensing experiment. The software module is used for the channel estimation and extracting the CSI values from the received signal. We conducted experiments to detect human presence and fall event detection. The experiments are conducted with the transmitter and receiver placed outside the wall of the lab, and the human activity (presence and fall) is undertaken in the lab as shown in Fig. 3. We also conducted the experiments with just one wall between the transmitter and receiver and no wall within the transmitter and receiver modules. The experiments are conducted with four different persons doing the activities. Each activity is repeated at least ten times, and the CSI values are captured for analysis. The capture CSI values usually contain noise due to environmental and hardware factors, and therefore, require preprocessing before analysis. There are various filtering techniques to filter out unwanted noise from the data.

We extracted the CSI features from the received signal, which contains both the amplitude and phase of the signal in different subcarriers. The CSI values are extracted as complex numbers, and the amplitude and phase part is separated using Algorithm 1. We applied different filtering techniques to remove the unwanted noise and smoothen the amplitude part of CSI data. We tried different filtering techniques like PCA, local outlier filter, Hampel filter and low pass filters. We implemented the Butterworth low pass filter as it seems to be working better in this application. The Butterworth low pass filter filtered out the noise very well compared to other filters.

After the data preprocessing step, we need to analyze the CSI amplitude data corresponding to frames and subcarriers. The CSI values contain 64 subcarriers,
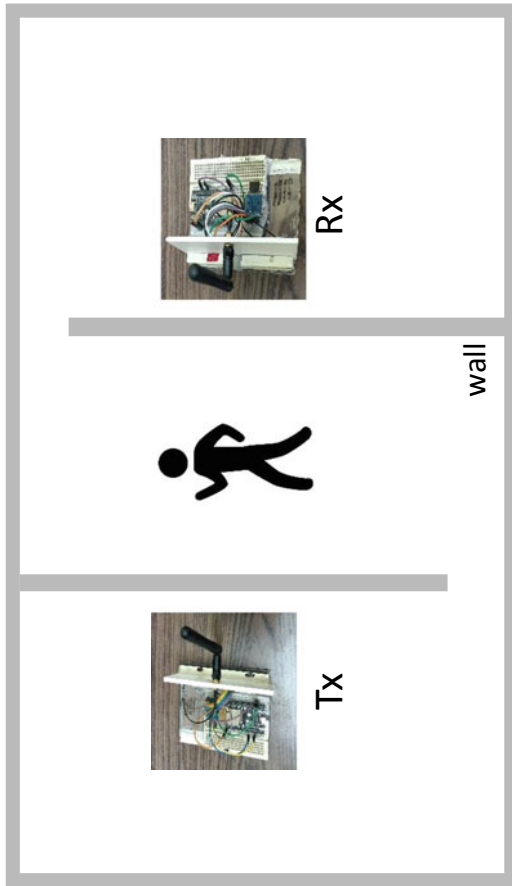
Fig. 3 Through-wall experiment of human activity

and patterns may be detected at any subcarriers. So, each activity is recorded for around 20 s, and about 800 frames are recorded per event. We trimmed some data frames from the start and end of the recorded measurement data to remove noise in the measurement setup operations. We adopted the sliding window technique with different window sizes to find the segment for detecting the event within collected data frames.

Algorithm 1 Channel state information data extraction algorithm

---

INPUT: Raw CSI data from the experiment
OUTPUT: Dataset consists of Amplitude ($\mathcal{A}$) and Phase (ϕ)
  $n \leftarrow$ number of frames or packets
  $m \leftarrow$ size of CSI data in each frame
  for $i = 1$ to n **do**
    for $j = 1$ to m **do**
      if $j\%2 == 0$ **then**
        $b \leftarrow Imaginary$ part of the CSI data
      else
        $a \leftarrow Real$ part of the CSI data
      end if
    end for
  end for
  for $k = 1$ to $m/2$ **do**
    Amplitude $\mathcal{A} \leftarrow \sqrt{a^2 + b^2}$
    Phase $\phi \leftarrow \arctan(\frac{b}{a})$
  end for
  Output: Amplitude [$\mathcal{A}_1, \mathcal{A}_2, \ldots \mathcal{A}_{64}$] and Phase [$\phi_1, \phi_2, \ldots \phi_{64}$]

---

5   Results and Analysis

Multiple experiments are conducted with no human presence, in the presence of a human, and with fall activities in LoS and NLoS scenarios. The CSI values are analyzed varying the window size between 10 and 20 frames to detect the events and accuracy is noted. We observe that window size 16 gives an optimal performance for the fall activity detection. Figure 4 shows the variations in the average amplitude of all the CSI subcarrier values corresponding to frames. A similar trend is observed in the average amplitude when the transmitter is placed without walls, with one wall in between and two walls separating them. Figure 5 shows the amplitude corresponding to the best subcarrier CSI variation when there is no wall between the transmitter and receiver. The green zone indicates no human activity/presence in the experiment zone. It can be observed that there is a significant change in the CSI amplitude when any activity is performed. Figure 6 shows the amplitude variation when the human presence and there is a fall event. The red colored zone in Fig. 6 shows the amplitude corresponding to the fall event, and the cyan colored zone shows only
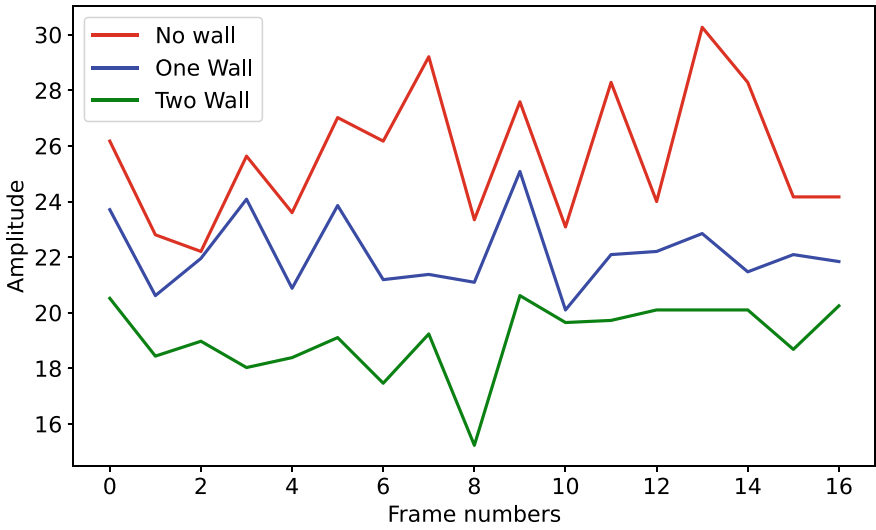
Fig. 4 Results of experiment with no wall, with one wall and two walls between the transmitter and receiver
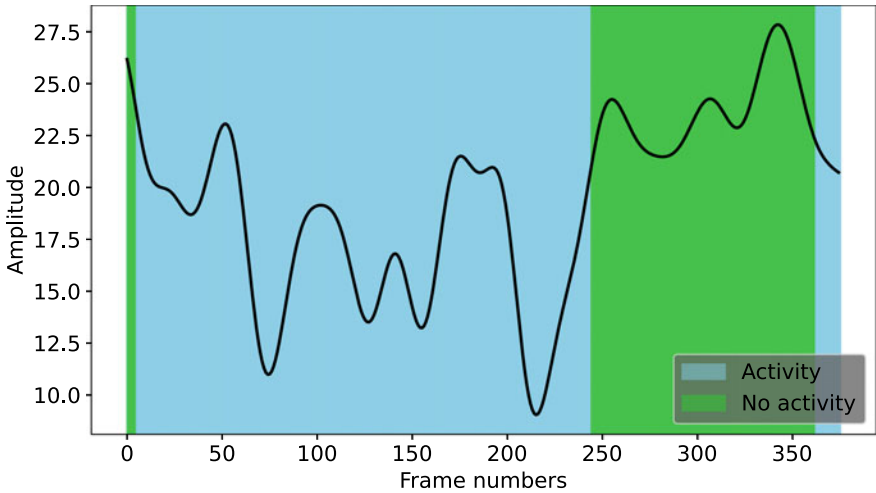
Fig. 5 Results of human activity with no wall between transmitter and receiver
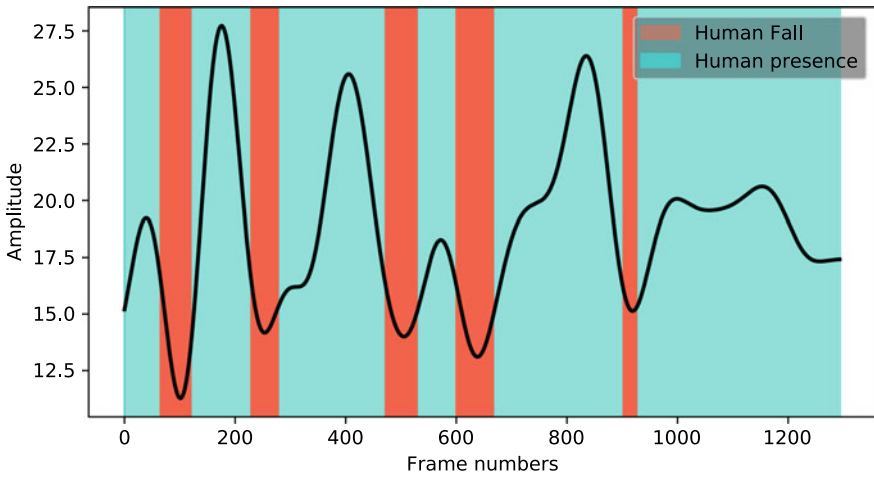
Fig. 6 Results of the through-wall sensing experiment when there is one wall between the transmitter and receiver
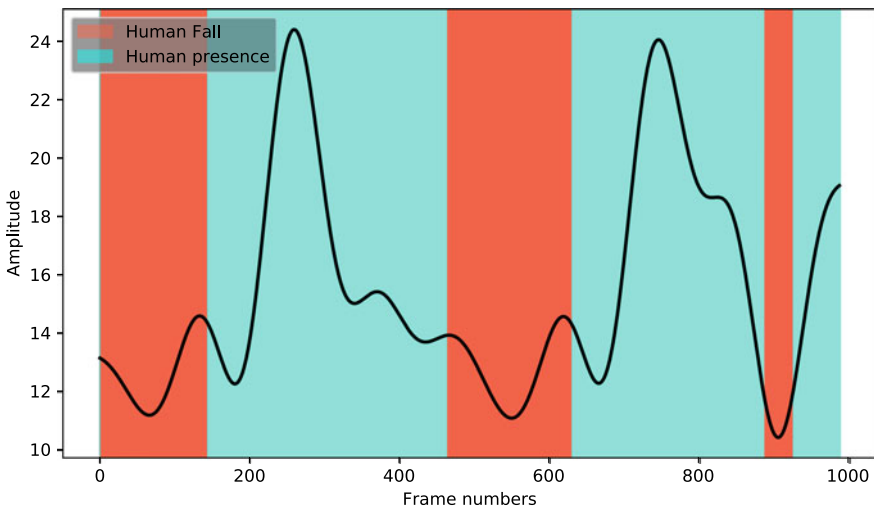
Fig. 7 Results of the through-wall sensing experiment when there are two walls between the transmitter and receiver

human presence. Thus, the variations in the amplitude can be used to detect the fall event. When the transmitter and receiver are separated with two walls, and the activity is conducted within the walls, the amplitude variations are shown in Fig. 7, which is similar to Fig. 6. The experimental results conclude that there is substantial variation in the received CSI values corresponding to different human activities, particularly

fall activity. The results and adjoining graphs show that the human presence and fall event can be easily detected using the amplitude part of the captured CSI values and appropriate segmentation techniques based on the sliding window method.

6 Conclusions and Future Scope

In this work, we demonstrate the use of Wi-Fi channel state information to sense through-wall human presence and fall event detection. We have shown the use of a low-cost ESP32 module for detecting the human presence and fall event detection. The preliminary results reported in this paper are pretty encouraging. We observed a clear variation even in the presence of obstacles like one wall between transmitter and receiver and two walls between the transmitter and receiver. This experiment has many potentials as it can detect human activity and fall events in a non-intrusive, noninvasive, device-free and non-cooperative environment. In the future, we would like to continue the experiments using low-cost devices like ESP32 to detect many other activities, gesture recognition, and build an end-to-end system to detect the events, which will include the use of different machine learning algorithms.

References

1. Ahmed HFT, Ahmad H, Aravind CV (2020) Device free human gesture recognition using Wi-Fi CSI: a survey. Eng Appl Artif Intell 87:103281. https://doi.org/10.1016/j.engappai.2019.103281
2. Atif M, Muralidharan S, Ko H, Yoo B (2020) Wi-ESP—a tool for CSI-based device-free Wi-Fi sensing (DFWS). J Comput Des Eng 7(5):644–656. https://doi.org/10.1093/jcde/qwaa048
3. Esp-idf programming guide (2020) https://docs.espressif.com/projects/esp-idf/en/latest/esp32/index.html. Accessed 30 July 2021
4. Esp32 hardware reference (2020) https://docs.espressif.com/projects/esp-idf/en/latest/esp32/hw-reference/index.html. Accessed 30 July 2021
5. Fang S, Alterovitz R, Nirjon S (2019) Non-line-of-sight around the corner human presence detection using commodity WiFi devices. In: Proceedings of the 1st ACM international workshop on device-free human sensing. DFHS'19. Association for Computing Machinery, New York, NY, pp 22–26. https://doi.org/10.1145/3360773.3360879
6. Gong L, Xiang C, Fan X, Wu T, Chen C, Yu M, Yang W (2020) Device-free near-field human sensing using WiFi signals. Pers Ubiquitous Comput. https://doi.org/10.1007/s00779-020-01385-4
7. Hao Z, Duan Y, Dang X, Zhang T (2020) CSI-HC: a WiFi-based indoor complex human motion recognition method. Mob Inf Syst 2020:1–20. https://doi.org/10.1155/2020/3185416
8. Hernandez SM, Bulut E (2020) Performing WiFi sensing with off-the-shelf smartphones. In: PerCom Demos 2020: 18th annual IEEE international conference on pervasive computing and communications demonstrations (PerCom Demos 2020), Austin, Mar 2020
9. Hernandez SM, Bulut E (2021) Adversarial occupancy monitoring using one-sided through-wall WiFi sensing. In: ICC 2021—IEEE international conference on communications, pp 1–6. https://doi.org/10.1109/ICC42927.2021.9500267
10. Li W, Tan B, Piechocki RJ (2018) WiFi-based passive sensing system for human presence and activity event classification. IET Wireless Sens Syst 8(7):276–283

11. Liu J, Teng G, Hong F (2020) Human activity sensing with wireless signals: a survey. Sensors 20(4). https://doi.org/10.3390/s20041210
12. Ma Y (2020) Improving WiFi sensing and networking with channel state information. Dissertations, theses, and masters projects, College of William & Mary, Williamsburg, VA, paper 1593091976
13. Ma Y, Zhou G, Wang S (2019) WiFi sensing with channel state information: a survey. ACM Comput Surv 52(3). https://doi.org/10.1145/3310194
14. Wang J, Gao Q, Pan M, Fang Y (2018) Device-free wireless sensing: challenges, opportunities, and applications. IEEE Netw 32(2):132–137. https://doi.org/10.1109/MNET.2017.1700133
15. Wang Z, Guo B, Yu Z, Zhou X (2018) Wi-Fi CSI-based behavior recognition: from signals and actions to activities. IEEE Commun Mag 56(5):109–115. https://doi.org/10.1109/MCOM.2018.1700144
16. Yang J, Zou H, Jiang H, Xie L (2018) Device-free occupant activity sensing using WiFi-enabled IoT devices for smart homes. IEEE Internet Things J 5(5):3991–4002. https://doi.org/10.1109/JIOT.2018.2849655
17. Yang J, Liu Y, Liu Z, Wu Y, Li T, Yang Y (2021) A framework for human activity recognition based on WiFi CSI signal enhancement. Int J Antennas Propag 2021:1–18. https://doi.org/10.1155/2021/6654752
18. Zhang R, Jing X, Wu S, Jiang C, Mu J, Yu FR (2021) Device-free wireless sensing for human detection: the deep learning perspective. IEEE Internet Things J 8(4):2517–2539. https://doi.org/10.1109/JIOT.2020.3024234

Design and Modeling of Virtual Robot for Industrial Application in Smart Manufacturing Assembly Line

D. Anil Kumar, Kali Charan Rath, Kamalakanta Muduli, and F. Ajesh

Abstract The robot is a machine and is a re-programmable, multi-functional regulator expected for transport various kinds of material, parts, instruments, or thought contraptions from end to end factor-altered developments for the carrying out of a grouping of tasks. Mechanical technology is portrayed as the discipline or examination of the development basically associated with the arrangement, production, speculation, and utilization of robots. Advanced mechanics are a science utilizing the proceeding with headways of mechanical design, sensor creation, material science, advanced calculations, and producing strategies, while various fields add to the science, the techniques, and the sections, mechanical independence makes the completed outcome. Robot makers can fabricate robots with more agreeable parts and mechanical constructions without losing execution, and robots can be utilized likewise in applications with extremely elite prerequisites, just like in smart manufacturing assembly line and machining. This paper presents the proposed and modeling of an industrial robot modeling in virtual platform to describe the end-effector trajectory through forward kinematics mechanism. Trajectory planning referred as path motion planning of end-effector. Now, end-effector is visiting through via points. It is observed from the simulation that the position and velocities are continuous by following C^0 continuity when end-effector passes through via points. Uniqueness of this work is that the trajectory produces smooth continuous motion while passing through assigned via points (generated intermediate points through iteration process algorithm).

Keywords Smart manufacturing system · Virtual robot · Robot simulation · Trajectory

D. Anil Kumar · K. C. Rath (✉)
GIET University, Gunupur, Odisha, India
e-mail: rathkalicharan1980@gmail.com

K. Muduli
Papua New Guinea University of Technology, Lae, Papua New Guinea

F. Ajesh
Sree Buddha College of Engineering, Alappuzha, Kerala, India

1 Introduction

The last century has seen a sensational change in the idea of work occurring in assembling enterprises. New systems, innovations, and practices are being presented on very nearly a consistent schedule, which makes the business among the most intriguing and testing. Maybe the most effective change has been the progress to computerization. While robotization has been around since the mid-twentieth century, it has procured gesture over the once decade or close, filled by the wide execution of cutting edge mechanics. The assembling business is going through a change. The assembling business has the chance to essentially reevaluate itself from effectiveness on the processing plant floor by accepting innovation. As interconnected and astute assembling is making progress, contending in the realm of Industry 4.0. In the manufacture of goods, robots and humans work together. The present innovation with shared advanced mechanics is one of the central participants in Industry 4.0.

The virtual robot is designed with the help of mechanical design software with proper dimension. Then, based on required degree of freedom, it is allowed for its smooth movement. The function of robot model is analyzed through virtual robot simulation platform with input parameter. The forward kinematics was tested in virtual mode by following DH rule. The transformation matrix of the simulation is equal with the analytical calculation value, and the trajectory is validated through end-effector traced path through polynomial function. Significance of the work is that the trajectory provides C^0 continuity with higher order polynomial up to certain degree of freedom only. It is observed that the continuity of the trajectory depends upon the number of intermediate points. As more points are associated in the trajectory, so the trajectory path traced by end-effector is smooth; otherwise, it follows C^1 continuity.

The goal of this paper is to give the peruser general machine in classified structure and calculation that can be applied together to tackle kinematics issues to a specific mechanical system according to necessity in sequential construction system. A simplest mathematical model executed through this paper to help the new researchers to design robot model. The robot body model is designed through design software. As per the features availability in the design software, the robot links get linked through joints with boundary condition values. Then, the simulation is applied to test the function of virtual robot. Positional change of end-effector and trajectory gets validated by free online robot simulation software. MATLAB features validate the interlinked virtual robot to evaluate the coordinate points visited by robot end-effector. The virtual robot model analysis is a cost-efficient testing process of any customized robot.

2 Literature Review and Proposed Methodology

The investigation of advanced mechanics offers exactly that chance to amalgamate space information going from mechanical and mathematics to hardware and software

engineering [1–3], as robots are mind boggling machines work with numerous layers of innovation [4–7].

As indicated by a new report, the advanced mechanics industry is ready to develop at a pace of in excess of 40% in the following decade, which is multiple times the development figures contrasted with the past couple of many years [8, 9]. Robots that are used in assembly line are more precision and task demanding [10]. As robot comes under mature technology, so it leads assembly line automation in various manufacturing industries in advanced technology [11–13]. It is assessed that with mechanical progressions, and robots will be a significant main thrust for building a manageable future [14, 15]. Industrial robots are driven by electronics and automotive industries to ensure high production at low costs [16, 17]. Indeed, even today, the term "robot" is utilized equivalently with the most recent popular expressions like mechanization, man-made consciousness, and large information, imagining a not so distant future where people and machines consistently collaborate to accomplish answers for true issues [18]. Use of robot in industry is now financially justified and has the ability to solve tight labor market [19, 20].

Kinematics centers around body development disregarding the powers or minutes that make that development. The expression "robot kinematics" alludes to the investigation of a robot regulator's development. For separating the conduct of current regulators, arranging sensible kinematics models for a robot instrument is very significant [21]. Kinematics of robot provides very accurate calculation, example as, positioning the gripper as per required position on the space, movement of tool from one point to the required operation position, and robot end-effector movement through trajectory [22]. Robot kinematics in geometry emphasis that the modeled robot ensures to provide rotation and translation as well [23]. Cartesian space and quaternion space are fundamentally two unique spaces utilized in kinematics demonstrating of controllers specifically [24]. From the study, it has been seen that the more research have to involve in kinematic analysis of robot. The simplest method is described in this paper to help new researchers to start work in robotics area.

3   Smart Manufacturing in Car Assembly Line

A smart factory permits the minimization of creation process times, just as its expenses, as it can adjust and upgrade the cycles as well as putting away and breaking down a lot of information continuously. Contrasted with people, robots move quicker and with more noteworthy precision. Diverged from individuals, robots move faster and with more imperative accuracy. They can assemble parts, embed screws and sticks, or apportion cements. Today, most robots are utilized in assembling tasks; the applications can be isolated into three classes: material taking care of, handling activities, and get together and investigation.

The provisions make robots the ideal associates for specialized or complex item mechanical production systems. More prominent exactness, expanded creation, and a pattern toward diminishing expense of the underlying cost are making robots in
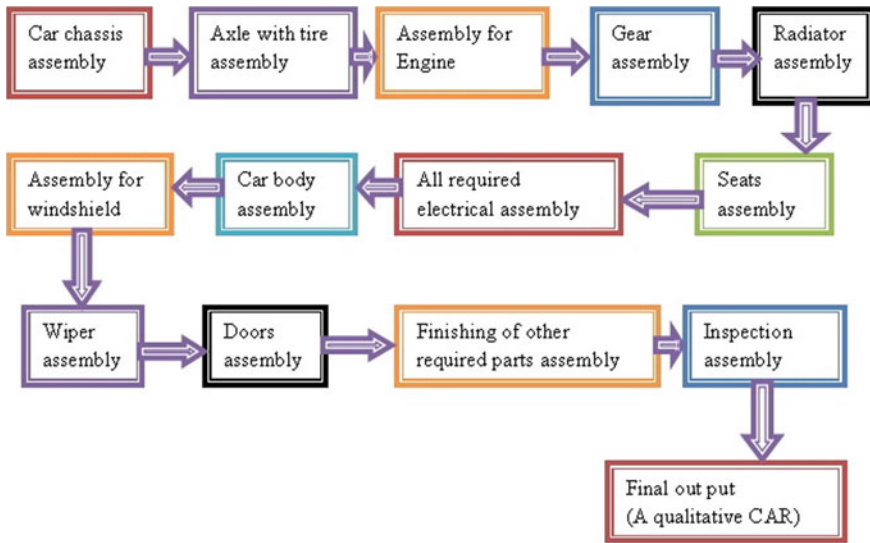
Fig. 1 Layout for car assembly line in smart manufacturing system

assembling substantially more alluring. Welding, painting, squeezing, gathering, pick and spot, palletizing, thing assessment, and testing are on the whole normal activities for ultramodern robots, which are all upgraded with high steadiness, speed, and delicacy. Robots that are used in quality control have the ability to check vehicle bodies for right shape, size, and estimation to ensure that later during creation parts will fit. General vehicle mechanical creation framework (Fig. 1) in sharp gathering floor is showed under.

3.1   Virtual Robot and Nomenclature

A robot is a machine worked to complete a perplexing undertaking (or set of errands) by truly moving and connecting with the workplace. A straightforward robot with three levels of opportunity can move in three ways: up and down, left and right, and forward and in reverse. The end-effector associates with the robot's arm and capacities as a hand. One of the principle parts of a robot is a fundamental part in any PC-driven innovation: the focal handling unit (CPU). The CPU goes about as the "cerebrum" of the robot. As such, a CPU is the robot part that gives input to outside improvements. The arm is the fundamental segment of the mechanical arm and comprises of three sections: the shoulder, elbow, and wrist. These are altogether joints, with the shoulder resting at the foundation of the arm, normally associated with the regulator, and it can push ahead, in reverse or twist. In like manner use,

"virtual" represents on the web that is through a PC or gadget. This is used to depict a correspondence that does not do in an in-person valid climate.

3.2 Design of Robot Manipulator Algorithm

Design of robot manipulator follows the following points:

Step 1: To finish the link coordinate frames, apply the DH rule.

Step 2: The manipulator's link-joint table is determined.

Step 3: DH transformation matrices are determined.

Step 4: Calculation of the joint coordinates as a function of the point P.

Step 5: Append the device facilities casing to P and address the forward kinematics to decide the casing's direction.

Step 6: Estimation for the remainder of the design and change framework of the controller (Fig. 2).

Design consideration for the model: (a) Three axes creates a spherical wrist on a six-axis arm, (b) from the central axis to the center of the wrist, the maximum reach is 880 mm, (c) payloads going from 5 to 2 kg are programming selectable, (d) estimated arm weight is 80 kg, (e) $\pm$ 0.1 mm as for repeatability, (f) straight line moves at 500 mm/sec with a maximum velocity of 2.5 kg, and (g) straight line moves at 470 mm/sec with a maximum velocity of 4.0 kg (Table 1).
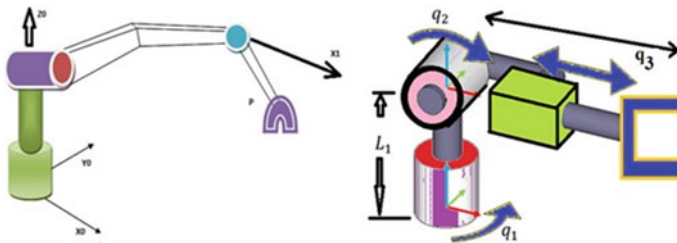
Fig. 2 Design of robot manipulator

Table 1 Model parameter value

Joint	Angle in degree	Joint	Angle in degree
waist	320	Wrist bend	200
Shoulder	266	Wrist roll	280
Elbow	284	Tool flange	532

3.3   Modeling for Forward Kinematics in Robot

Kinematics is the investigation of the connection between a robot's joint directions and its spatial format. Forward kinematics utilizes a robot's kinematic states to select the end-circumstance effector's from known traits for joint impediments. The volume of space that the robot can reach, known as its work area, is defined by the robot's components and kinematics circumstances. A kinematic model portrays the development of a robot in mathematical construction dismissing the forces that impact development and stresses over the numerical association between parts. Kinematic showing of regulators expects a critical part in contemporary robot control. It depicts the association between robot end-effector position and bearing in space and regulator joint focuses, as shown in Fig. 3a–c.

Link parameters	Joint parameter
Link length	Distance between links
Link twist	Angle between links
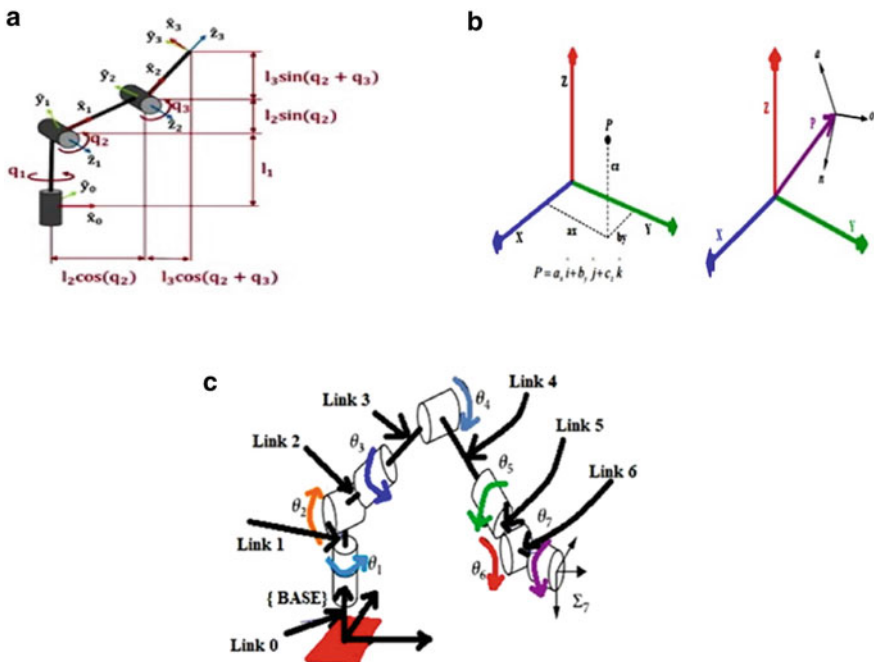
Fig. 3 **a** Geometry for forward kinematics of a three DoF robot manipulator, **b** point in space, and **c** link and joint mechanism
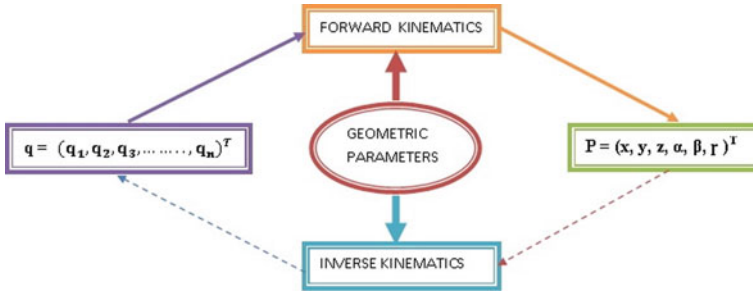
Fig. 4 I/P and O/P parameter for kinematics mechanism of robot

$$F = \begin{vmatrix} n_x & O_x & a_x & P_x \\ n_y & O_y & a_y & P_y \\ n_z & O_z & a_x & P_z \\ 0 & 0 & 0 & 1 \end{vmatrix} \tag{1}$$

From the arm's foundation to the last moving link, the connection numbering is displayed as Fig. 4.

Showing the robot's computation is tended to in kinematics. In most kinematics models, homogenous transformation is used. Given known joint angles and angular velocities, forward kinematics refers to the interaction of obtaining end-effector position and speed. For example, in the event that the arm's shoulder and elbow joint focuses are given in the sagittal plane, the goal is to choose the Cartesian headings of the wrist/grasp hand.

The kinematic state of manipulator is typically defined with joint angles and angular velocities.

$$\theta_1, \theta_2 = X_{EE}, y_{EE} \tag{2}$$

When joint-1 is located at origin, the coordinate of the joint-2 is:

$$X_2 = L_1 \cos(\theta_1) \tag{3}$$

$$Y_2 = L_1 \sin(\theta_1) \tag{4}$$

Now, coordinates of end-effector can be calculated with the help of following mathematical model.

$$X_{EE} = X_2 + L_2 \cos(\theta_1 + \theta_2) = L_1 \cos(\theta_1) + L_2 \cos(\theta_1 + \theta_2) \tag{5}$$

$$Y_{EE} = Y_2 + L_2 \sin(\theta_1 + \theta_2) = L_1 \sin(\theta_1) + L_2 \sin(\theta_1 + \theta_2) \tag{6}$$

Derivative of the above equations will provide the end-effector velocity

$$\dot{X}_{EE} = -\dot{\theta}_1 L_1 \sin(\theta_1) - (\dot{\theta}_1 + \dot{\theta}_2) L_2 \sin(\theta_1 + \theta_2) \tag{7}$$

$$\dot{Y}_{EE} = \dot{\theta}_1 L_1 \cos(\theta_1) + (\dot{\theta}_1 + \dot{\theta}_2) L_2 \cos(\theta_1 + \theta_2) \tag{8}$$

3.4 Trajectory Generation and Kinematic Constraints

The approach for generating trajectories is based on satisfying velocity, acceleration, and jerk restrictions. All through the proposed movement, every constraint should stay steady. Every hub can have different imperatives in the multidimensional situation. The joint-space heading time occurs at runtime for all n joints openly and simultaneously. Way-points, that are a series of points along the path, are commonly used to illustrate the planned path. Course age is just it is making a bearing interfacing no less than two-way centers. Each through point has the chance that the robot end-effector visits.

Position of end-effector

$$P(t) = (x(t), y(t), z(t)) \tag{9}$$

Velocity

$$V(t) = \dot{P}(t) = (x'(t), y'(t), z'(t)) \tag{10}$$

Acceleration

$$a(t) = \ddot{P}(t) = (x''(t), y''(t), z''(t)) \tag{11}$$

Both the joint space and the functional space can be utilized to create directions. Two separate control approaches are used to govern trajectory: (a) displacement control and (b) force control. Controlling the end-effector: Polynomials are one of the most common types of functions associated (cubic, quintic, sinusoids, clothoids, etc.). The following are the conditions that must be met:

1. Interpolation, which is the process of moving from one location to another.
2. The path's initial, final, intermediate, or geometric tangent velocity.
3. Geometric curvature or initial, ultimate, or intermediate velocity
4. Continuity up to C^k.

4 Forward Kinematic Analysis of Virtual Robot Puma 560

Forward kinematics demonstrates joint situations, for instance the marks of the revolute joints and joint movements. The purpose is to figure out the end-posture, effector's, or its location and orientation. This can be refined utilizing the conclusion conditions depicted in the stages underneath.

Connect an organize casing to each of the $(n + 1)$ connections of a chronic controller with n levels of opportunity, with Frame 1 connected to the fixed or base casing and Frame $(n + 1)$ to the furthest limit of the nth connection. DH boundaries are clear cut. Homogeneous change grids were determined as $T_1, T_2, T_3, …, T_i, …, T_n$. The homogeneous change lattice of the end-effector outline regarding the Frame 1, i.e., T calculated as $T = T_1, T_2, T_3, …, T_i, …, T_n$.

The direction of the bodies or interfaces and the places of the starting points of the casings joined to the connections were gotten as follows:

$$Q = Q_1, Q_2, Q_3, …, Q_i, …, Q_n \tag{12}$$

$$P = a_1 + Q_1 a_2 + \cdots + Q_i a_i + \cdots + Q_{n-1} a_n \tag{13}$$

where Q_i is the symmetrical revolution grid which addresses the direction of casing—$(i + 1)$ concerning the casing—i and vector a_i is the vector from the beginning of frame i to the beginning of edge $(i + 1)$. So for the given sort of joint for example revolute or diverse one of as far as possible is variable which is known as the joint variable, but the other three excess cutoff points are unsurprising and are called interface limits (Figs. 5 and 6).

Output Result

The end-effector positional output for virtual robot PUMA 560. During this simulation, except joint-1 parameter, other joint parameter values were taken as fixed. Only joint-1 parameter value changed to observe the trajectory of end-effector. Incremental

Angle	Range	DoF
T_1	-160 to +160	Waist Joint
T_2	-225 to 45	Shoulder Joint
T_3	-45 to 225	Elbow Joint
T_4	-110 to 170	Wrist Roll
T_5	-100 to 100	Wrist Bend
T_6	-266 to 266	Wrist Swivel
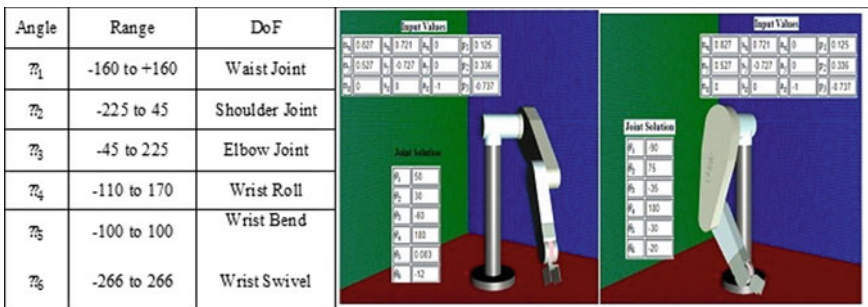
Fig. 5 Input joint angle range
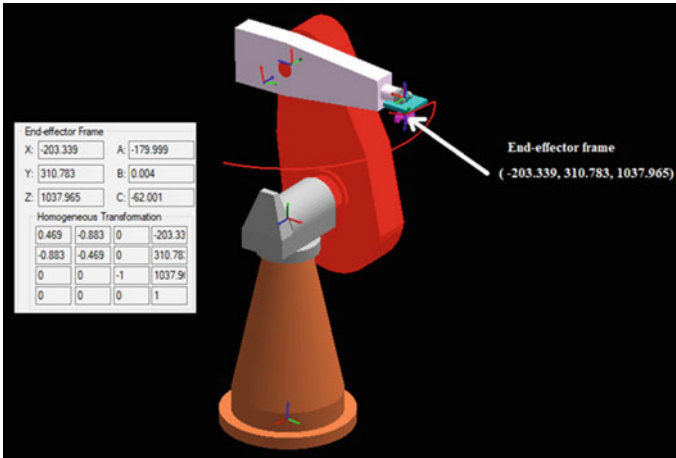
Fig. 6 Homogeneous transformation, end-effect frame, and position

position provided through 1 mm and increment in joint angle provided 0.5°. Coordinate of the end-effector position has been recorded through each increment of 1° in θ_1 (Fig. 7; Tables 2 and 3).

The homogeneous transformation matrix table at the target position (-457.25, 6.789, and 1038.00) is.
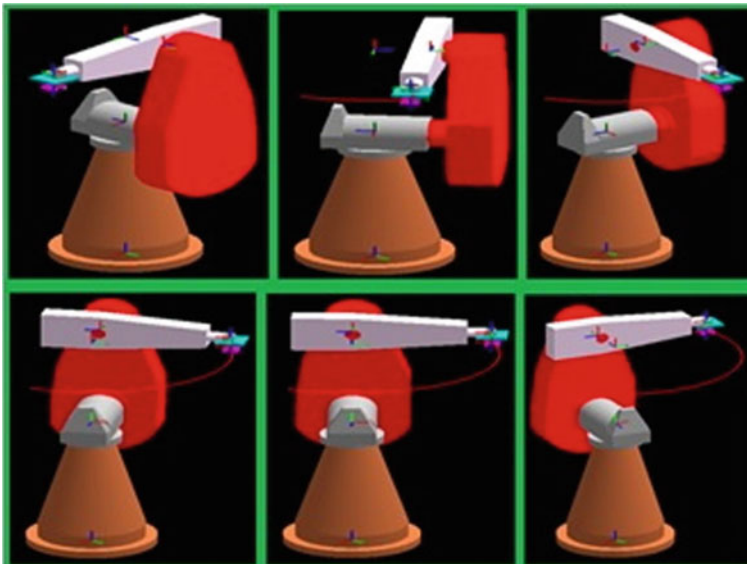
Fig. 7 Trajectory drawn by end-effector C^0-continuity

Table 2 Joint angle parameter

Joint parameter	Value in degree	Joint parameter	Value in degree
θ_1	0°–160°	θ_4	0°
θ_2	90°	θ_5	−270°
θ_3	0°	θ_6	0°

Table 3 Positional coordinate value of end-effector of robot PUMA 560

Joint parameter θ_1 (in degree)	Coordinate of end-effector		
	Xi	Yi	Zi
0°	432	150	1038
1°	429.289	157.53	1038
2°	426.506	164.987	1038
…			
139°	−424.445	170.23	1038
…			
160°	−457.25	6.789	1038

0.94	−0.342	0	−457.25
−0.342	−0.94	0	6.789
0	0	−1	1038
0	0	0	1

The connection between the interesting joints of the robot controller and the area and bearing of the hardware or end-effector is the focal point of forward kinematics. Forward kinematics, to put it all the more officially, decides the area and bearing of the end-effector dependent on the properties of the robot's joint parts. This virtual assessment portrays a forward kinematic study to decide the general effect of every single joint variable. The trajectory traced by end-effector follows cubic polynomial with $C°$ continuity.

5 Conclusion

The bend parameterization in time $q = q(t)$ is adequate to work part insightful, i.e., q_i in the vector q. Yet, a certain meaning of the direction is gotten by tackling the planned issue with determined limit conditions in a given class of capacities as it were. The kinematics of a robot relate the joint points of a robot to the arrange edges of its connections. A robot's reachable work area is the scope of end-effector areas it could reach. Forward kinematics figures the arrange outlines comparing to

robot's setup. Estimating the situation of an arm's end-effector is just a question of understanding the kinematics of the robot and estimating the situation of every single halfway joint. The investigation of the arrangement of staggered of chance kinematic chains that shape the creation of mechanized structures is called robot kinematics. The emphasis on calculation means that the robot's connections are represented as rigid bodies, and its joints are required to provide unbiased pivot or interpretation. Validation of work is proposed through the virtual simulation mode and found that the end-effector moves in smooth trajectory with C^0 continuity.

Further work of this examination centers around effectively applying various portrayals of position, direction, and their subsidiaries as for time to take care of different kinematics issues as needed to connect with workplace in industry.

References

1. White T, Alexander R, Sargent J (2005) A mobile climbing robot for high-precision manufacture and inspection of aero-structures. In: 5th International conference on climbing and walking robots (CLAWAR'03), and Catania (Italy), vol 24, pp 589–598
2. Arnay R, Hernández-Aceituno J, González E, Acosta L (2017) Teaching kinematics with interactive schematics and 3D models. Comput Appl Eng Educ 25:420–429
3. Bulanon DM, Kataoka T, Okamoto H, Hata (2005) Feedback control of manipulator using machine vision for robotic apple harvesting. In: Proceedings of ASAE, Paper No. 053114, Tampa, USA, pp 531–539
4. Chun Y, Bidanda B (2013) Sustainable Manufacturing and the Role of the International Journal of Production Research. Int J Prod Res 51:23–24
5. Colleoni E, Edwards P, Stoyanov D (2020) Synthetic and real inputs for tool segmentation in robotic surgery. In: International conference on medical image computing and computer-assisted intervention. Springer, Cham, pp 700–710
6. Fomin A, Antonov A, Glazunov V (2020) Forward kinematic analysis of a rotary hexapod. In: Symposium on robot design, dynamics and control, proceedings of the 23rd CISM IFToMM symposium. Springer, Cham, pp 486–494
7. Gao R (2020) Inverse kinematics solution of Robotics based on neural network algorithms. J Ambient Intell Humaniz Comput 11(3):199–209
8. Jiang X, Chen X, Yu Z, Zhang W, Meng L, Huang Q (2018) Motion planning for bipedal robot to perform jump maneuver. Appl Sci 8(1):139
9. Kennedy B, Aghazarian H, Cheng Y, Garrett M, Hickey G, Huntsberger T, Magnom L, Mahoney C, Meyer A, Knight J (2001) Lemur: legged excursion mechanical utility rover. Autonomous Rob 11:201–205. https://www-robotics.jpl.nasa.gov/publications/Terrance_Huntsberger/LEMUR_AR.pdf
10. Katypitiya J, Eaton R, Cole A, Meyer C, Rodnay G (2005) Automation of an agricultural tractor for fruit picking. In: IEEE international conference on robotics and automation, Barcelona, Spain, pp 3201–3206
11. Kong X, Gosselin CM (2005) Type synthesis of 5-DOF parallel manipulators based on screw theory. J Robot Syst 22(10):535–547
12. Loreto-Gomez G, Rodriguez-Arce J, Gonzalez-Garcia S, Montano-Serrano VM (2018) Analysing the efect of the use of 3D simulations on the performance of engineering students in a robotics course: fndings from a pilot study. Int J Electr Eng Educ 002072:091879011
13. Loreto-Gomez G, Rodriguez-Arce J, Gonzalez-Garcia S, Montano-Serrano VM (2018) 8: Analysing the efect of the use of 3D simulations on the performance of engineering students in a robotics course: fndings from a pilot study. Int J Electr Eng Educ 002072:091879011

14. Mateus D, Avi~na G, Devy M (2005) Robot visual navigation in semi-structured outdoor environments. In: IEEE international conference on robotics and automation, Barcelona, Spain, pp 4702–4707
15. Macke N, Rulhoff S, Stjepandic J (2016) Advances in smart manufacturing change management. In: Transdisciplinary engineering: crossing boundaries. IOS Press, pp 318–328. https://doi.org/10.3233/978-1-61499-703-0-318
16. Moon JY, Park J (2014) Smart production scheduling with time-dependent and machine-dependent electricity cost by considering distributed energy resources and energy storage. Int J Prod Res 52(13):3922–3939
17. Nubiola A, Bonev IA (2013) Absolute calibration of an ABB IRB 1600 robot using a laser tracker. Robot Comput Integr Manuf 29(1):236–245
18. Pott A, Franitza D, Hiller M (2004) Orientation workspace verification for parallel kinematic machines with constant leg length. In: Proceedings of mechatronics and robotics, Aachen, Germany, pp 984–989
19. Pott A (2018) Cable-driven parallel robots—theory and application. Springer, Cham, Springer Tracts in Advanced Robotics
20. Park SO, Lee MC, Kim J (2020) Trajectory planning with collision avoidance for redundant robots using jacobian and artificial potential field-based real-time inverse kinematics. Int J Control Autom Syst 18(8):2095–2107
21. Ryu SW, Lee JG, Kim HJ (2020) Design, fabrication, and analysis of flapping and folding wing mechanism for a robotic bird. J Bionic Eng 17(2):229–240
22. Thoben K-D, Wiesner S, Wuest T (2017) Industry 4.0 and smart manufacturing—a review of research issues and application examples. Int J Autom Technol 11(1):4–16
23. Zhong RY, Xu C, Chen C, Huang GQ (2017) Big data analytics for physical internet-based intelligent manufacturing shop floors. Int J Prod Res 55(9):2610–2621
24. Zhong B, Xia L (2020) A systematic review on exploring the potential of educational robotics in mathematics education. Int J Sci Math Educ 18(1):79–101

A Hybrid CNN Real-Time Object Identification and Classification Approach for Autonomous Vehicles

Sanjoy Choudhury, B. H. Karthik Pai, K. Hemant Kumar Reddy, and Diptendu S. Roy

Abstract Autonomous Vehicles (AVs) are progressively attaining attention globally. It consists of clear and transparent potential future in the upcoming generations and it is estimated that it will intensely differ the transportation as we know it by today. An automated driving system is a complicated and complex association of several constituents. The identification and classification of any object and traffic road sign is an important job for any autonomous vehicle at the same time achieving the 100% accuracy is also difficult and complex task. In this paper, we have put forward the proposal for hybrid CNN deep learning model for real-time identification and classification of objects and simulation-based results proved that the modified version CNN model performed very well with accuracy more than 95%.

Keywords Autonomous driving · Convolutional neural networks · Object detection · Deep learning

S. Choudhury (✉) · D. S. Roy
National Institute of Technology, Meghalaya, Shillong, India
e-mail: sanjoy.choudhury@bose.res.in; p19cs009@nitm.ac.in

D. S. Roy
e-mail: diptendu.sr@nitm.ac.in

S. Choudhury
S. N. Bose National Centre for Basic Sciences, Kolkata, India

B. H. Karthik Pai · K. Hemant Kumar Reddy
NITTE, Mangaluru, India
e-mail: karthikpai@nitte.edu.in

K. Hemant Kumar Reddy
e-mail: khemant.reddy@nitte.edu.in

1 Introduction

One of the most awaited smart city services are intelligent transportation systems. Transportation infrastructure with connectivity, autonomy, sensing and the endowing vehicles are the foundation on which ITSs is built upon in order to guarantee effective transportation and provide harmless road travel [1]. To facilitate this ITS vision, furnishing of transportation infrastructure and vehicles along with smart sensor need to be done which can collect and process bulk of heterogeneous data on individual passengers, vehicle and its environment. Real time and ultra-low latency collection [2] of information must be completed so to provision various autonomous features including self-driving vehicles by ITSs. Certainly, data analytics infrastructure and reliable communication to process and transmit the data generated by the sensors is required for managing and monitoring the several ITS operations. However, because of enormous number of mobile sensors in an ITS, communication of all the measurements of sensors to a cloud that is located remotely by using traditional method [3] can result in computational overload, communication network congestions and high delays which is intolerable for any ITS, as any delayed decision in the system potentially can cause accidents, travel delays and congestions.

1.1 Convolution Neural Networks

From past few decades, several neural network frameworks have been established to resolve different tasks. The most upfront way to build a neural network is the Feedforward neural networks in which the data flows from the input neurons in unidirectional way to the output neurons. The Single Shot MultiBox Detector (SSD) is one of the famous architectures [4], which act as the basic architecture.

1.2 Convolutional Neural Networks Models

With an aim to understand and sense our world, A convolutional neural network (CNN) is a form of artificial neural network that is especially intended to analyse pixel input and is used in image recognition and processing. A neural network is a hardware and/or software system modelled is one of the most challenging study topics in today's smart computing. Many of the latest and efficient research in the field of green computing is based on the well-known CNNs architectures.

LeNet   One of the earliest applications of Convolutional Networks which was successfully corresponding to computer vision problems is LeNet, it was introduced in 1990s by Yann LeCun. The LeNet model which is extensively utilized for classification problems.

AlexNet In Computer Visions, the actual breakthrough of Convolutional Networks was symbolized by AlexNet which was implemented by Krizhevsky et al. [5]. The basic architecture of AlexNet was much alike to the LeNet, but it was much bigger and deeper.

GoogLeNet The next ILSVRC 2014 challenge on object detected was achieved by GoogLeNet, which is an additional remarkable CNN architecture in terms of inception [3]. It achieved an unbelievable top-5 accuracy of 93.33%. Advancement of the Inception Modules is one of its significant contributions which was motivated to various authors.

VGGNet One of the most homogeneous, deep and simple structures in CNN is represented by the VGGNet. VGGNet was proposed by Simonyan and Zisserman in [6]. It achieved the second-best stance in the ILSVRC 2014 challenge. The main target of VGGNet was to demonstrate the importance of depth in terms of performance in CNN.

ResNet by Kaiming et al. invented the Residual Network. He won the ILSVRC 2015 award. ResNet makes use of hefty batch normalization and thus contains important skip connections features. CNNs consist of deep architectures which require a plenty of memory causing difficulties in dealing with GPUs.

The following are some of the study's major contributions we have put forward for new directions of smart systems presented in this paper:

1. Three architectures have been defined and used for comparisons including simple, modified LeNet and VGGNet consisting of 8 layers.
2. We employed the YOLO model to enable end-to-end training and real-time speeds while maintaining high average precision, keeping in mind the key problem of memory utilization and speed.
3. Computational simulations are conducted and furnished in order to the reliability and efficacy of the proposed model.

The remainder of this paper is structured in the following manner. In Sect. 2, the relevant works are discussed. Architectures including simple, modified LeNet and VGGNet consisting of 8 layers and its implementation as well as it presents the benefit and usage of YOLO are discussed in Sect. 3. In Sect. 4, extensive experiment findings are presented accompanied by concluding remarks in Sect. 5.

2 Related Work

The plausible improvement in the safety and performance of vehicle, its adequacy in traffic [4] along with save in energy [7] have led to significant growth to the autonomous vehicles and thus most of the technology trends are inclined towards automotive industry in the past decades. Industry and academia both have already attracted a significant attention to research in the field of automotive industry. Rang-

ing from high-density flow of traffic to the highways with high-speed, and in various urban to suburb scenarios, there must be safe and adequate travelling trajectories, for which the autonomous vehicles must be spontaneous, precise, stable and computationally efficient. There are numerous difficulties and ambiguities surrounding the road and weather circumstances as well as the traffic in real-world also consists of dynamic. Presently, based upon the presence of two research methods, the research on autonomous vehicle's object classification can be carried upon in different ways. The first type of research includes applications based upon RGB-D. Imran et al. used Kinect to produce depth image, later combining it with RGB image, and using four-channel data flow, the convolution network was trained [6, 8].

OverFeat [1] is a Convolutional Neural Network method that does pattern recognition, detection, and localization all at the same time. OverFeat is one of the most effective detection approaches up to date, having won the ImageNet Large Scale Visual Recognition Challenge 2013 job of stratification.

VGG16 [8] attempted to overtake OverFeat's dominance in object categorization and detection by investigating the implications of excessive layer depth. In the ImageNet ILSVRC 2014 Challenge, layers proposal were created, establishing up to date region models in localization and stratification.

YOLO [9] is a new model that works with images directly and treats object recognition as regression rather than classification. YOLO has come together on CNN based object detection model, proposed by Joseph et al. in 2016 [9]. Benefitting from the unified design, YOLO beats the other state-of-the-art methods as its detection speed is 10 times faster [9].

Fast R-CNN [10] aims for predicts the accuracy of depth models while speeding them up. Fast R-CNN assumes on the region models and uses shared resources per region models and reduced SVD factorizations to improve speed and model accuracy time.

Hence, by considering the above challenges in mind and the importance of Identification and classification of any object and traffic road for any autonomous vehicle at the same time achieving the 100% accuracy is also difficult and complex task. Our major goal is to offer a hybrid CNN deep learning model for real-time item recognition and classification utilizing YOLO.

3 System Model and Architecture Design

Our neural network is based on YOLO model. However, in this section, we will explain the three different CNN architecture which is used for modelling, implementation and comparisons. Simple CNN architecture is that this architecture only

Table 1 Modified version of LeNet architecture consisting of 8 layers

S. No.	Parameters	Description
1	Input	Image of 32×32 size with 1 channel (Grayscale)
2	Layer 1 (Convolutional)	6 filters of 5×5 size
3	Activation	ReLu activation function
4	Pooling	2×2 filter with stride 2
5	Layer 2 (Convolutional)	16 filters of 5×5 size
6	Activation	ReLu activation function
7	Pooling	2×2 filter with stride 2
8	Layer 5 (Fully Connected)	It should have 120 outputs
9	Activation	ReLu activation function
10	Layer 4 (Fully Connected)	It should have 43 outputs, Softmax Regression to find probability of all classes
11	Optimizer	Adam Optimizer
12	Loss function	Categorical Cross entropy

helps to find digit but not complex patterns. Table 1 represents the modified version of LeNet. Table 2 presents the modified version of VGGNet consisting of 8 CONV/FC layers.

3.1 Problem Formulation

Entire image is used to extract features such that each bounding box can be predicted by the network. At the same time, it also does prediction of all classes' all bounding boxes. In addition to the ability to maintain high average accuracy, the design of YOLO also enables speed in real-time along with end-to-end training. There is a division of input image into the grid of $S \times S$, by the system.

1. Confidence Score
   The confident score is the measure of confidence of model to contain an object in the box, as well as if the box that it predicts, is precise enough or not. The confidence is formally described in Eq. 1. There must be a zero-confidence score in case of absence of any object in that cell. In all other cases, the intersection over union (IOU) between the ground truth and the predicted box makes up the confidence score, as represented in Eq. 2.

$$\text{Probability(Object)} \times \text{(IOU)truthpred} \qquad (1)$$

Table 2 Modified version of VGGNet architecture consisting of 8 layers

S. No.	Parameters	Description
1	Input	Image of 50×50 size with 3 channels (R, G, B)
2	Layer 1 (Convolutional)	The shape of output must be $50 \times 50 \times 32$
3	Activation	ReLu activation function
4	Layer 2 (Convolutional)	The shape of output must be $50 \times 50 \times 32$
5	Activation	ReLu activation function
6	Pooling	The shape of output must be $25 \times 25 \times 32$
7	Layer 3 (Convolutional)	The shape of output must be $25 \times 25 \times 64$
8	Activation	ReLu activation function
9	Layer 4 (Convolutional)	The shape of output must be $25 \times 25 \times 64$
10	Activation	ReLu activation function
11	Pooling	The shape of output must be $12 \times 12 \times 64$
12	Layer 5 (Convolutional)	The shape of output must be $25 \times 25 \times 128$
13	Activation	ReLu activation function
14	Layer 6 (Convolutional)	The shape of output must be $25 \times 25 \times 128$
15	Activation	ReLu activation function
16	Pooling	The shape of output must be $6 \times 6 \times 128$
17	Flattening	Final pooling layer's output shape must be flattened such that it's 1D instead of 3D
18	Layer 7 (Fully Connected)	This should have 128 outputs
19	Activation	ReLu activation function
20	Layer 8 (Fully Connected)	This should have 43 outputs, Softmax Regression to find probability of all classes
21	Optimizer	Adam Optimizer
22	Loss function	Categorical Cross entropy

$$\text{Confidence Score } C = \begin{cases} 0, & \text{if object} = 0 \\ \text{IOU}, & \text{if object} \neq 0 \end{cases} \qquad (2)$$

There exist five predictions, namely, x, y, w, h and confidence, for every single bounding box. With respect to the grid cells' bounds, the box's centre is symbolized by the (x, y) coordinates. The prediction of height and width takes place with respect to the whole image. Lastly, the IOU among the ground truth box and the predicted box is symbolized by confidence prediction. In accordance with Eq. 3, there is prediction of C conditional class probabilities by every single grid cell. Regardless of the prediction of B number of boxes, only a single set of class probabilities per grid cell is chosen.

$$\text{Probability}(\text{Class}(i)|\text{Object}) \qquad (3)$$

Often, there is a multiplication of individual box confidence predictions with the conditional class probabilities, thus resulting into class-specific confidence scores for each box. There is encoding of both the class' probability of being present in the box, as well as the wellness of fitting of object in the predicted box, as represented in Eq. 4.

$$\Pr(\text{Class}(i)|\text{Object}) \times \Pr(\text{Object}) \times (\text{IOU})\text{truthpred} \\ = \Pr(\text{Class}(i)) \times (\text{IOU})\text{truthpred} \tag{4}$$

2. Linear Activation Function

In accordance with Eq. 5, it depicts that the following leaky rectified linear activation is being used by the final and all other layers of the linear activation function.

$$\Phi(x) = \begin{cases} x, & \text{if } x > 0 \\ 0.1x, & \text{otherwise} \end{cases} \tag{5}$$

3. Loss Function

The various loss functions used are:

(1) *Classification loss:*

As represented in Eq. 6, for each class, each cell's classification loss is the squared error of the class conditional probabilities, in case of detection of an object.

$$\sum_{i=0}^{S^2} 1_i^{\text{obj}} \sum_{C \in \text{classes}} (p_i(c) - \hat{p}_i(c)^2) \tag{6}$$

$1_i^{\text{obj}} = 1$ if an object emerges in ith cell, otherwise 0.

$\hat{p}_i(c)$ implies that the cell i for class c has conditional class probability.

(2) *Localization loss:*

For the prediction of errors in the boundary box locations and sizes, the localization loss is used. As represented in Eq. 7, the box accountable for object's detection, is only counted.

$$\lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right] + \\ \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right] \tag{7}$$

$1_{ij}^{\text{obj}} = 1$ if object is detected by cell i of the jth bounding box, otherwise 0.

λ_{coord} increases the capacity of the loss in the coordinates of bounding box. There must not be equal weight assigned by absolute errors in small and

large boxes, that is, there must not be same 2-pixel error in a large as well as a small box. So, the model followed by YOLO is to predict the square root of the bounding box height and width rather than directly taking height and width into consideration. Additionally, to emphasize more on precision of the bounding box, the loss is simple multiplied by $\lambda_{\text{coord}} = 5$.

(3) *Confidence loss:*
Equation 8 represents the formula of confidence loss when there is both conditions of detection as well as non-detection of objects within the box.

$$
\text{Confidence loss} = \begin{cases} \sum\limits_{i=0}^{S^2} \sum\limits_{j=0}^{B} 1_{ij}^{\text{obj}} (C_i - \hat{C}_i)^2, & \text{if object} \neq 0 \\ \lambda_{\text{noobj}} \sum\limits_{i=0}^{S^2} \sum\limits_{j=0}^{B} 1_{ij}^{\text{noobj}} (C_i - \hat{C}_i)^2, & \text{if object} = 0 \end{cases} \tag{8}
$$

$1_{ij}^{\text{obj}} = 1$ if object is detected by the cell i of the jth bounding box, otherwise 0, C_i is the box confidence score of the box j in cell i. λ_{noobj} weights down the loss when detecting background. 1_{ij}^{noobj} is the complement of 1_{ij}^{obj}.
Due to absence of objects in most of the boxes, there occurs a class imbalance problem, under which the model is trained such that there is more frequent detection of background than the detection of objects, to nullify which, we reduce the weight by a factor $\lambda_{\text{noobj}} = 0.5$.

(4) *D. Loss:*
The final loss is represented by the addition of localization, confidence and classification losses together as given in Eq. 9.

$$
\text{Final Loss} = (6) + (7) + (8) \tag{9}
$$

By using Eqs. 5–9 a hybrid architecture is formed which is different from the standard CCN models.

4   Performance Evaluation

In this section, we present the findings of our simulation-based tests, which can be used to assess the effectiveness and efficiency of our suggested approach. The KITTI Object Detection Benchmark 2012 [1] dataset is used in this study.

We used an Amazon AWS g2.2xlarge server running Ubuntu 16.04 for our computing environment; this environment was useful because it is inexpensive, configurable, and includes GPU hardware with pre-configured NVIDIA drivers. An NVIDIA GT920M graphics card, 1 TB of hard disc capacity, 16 GB of RAM, and a 7-core CPU are among the machine's features.

```
Layer (type)                    Output Shape              Param #
================================================================
conv2d_1 (Conv2D)               (None, 32, 28, 28)        832
----------------------------------------------------------------
max_pooling2d_1 (MaxPooling2    (None, 32, 14, 14)        0
----------------------------------------------------------------
conv2d_2 (Conv2D)               (None, 64, 14, 14)        51264
----------------------------------------------------------------
max_pooling2d_2 (MaxPooling2    (None, 64, 7, 7)          0
----------------------------------------------------------------
flatten_1 (Flatten)             (None, 3136)              0
----------------------------------------------------------------
dense_1 (Dense)                 (None, 1024)              3212288
----------------------------------------------------------------
dropout_1 (Dropout)             (None, 1024)              0
----------------------------------------------------------------
dense_2 (Dense)                 (None, 43)                44075
================================================================
Total params: 3,308,459
Trainable params: 3,308,459
Non-trainable params: 0
```

Fig. 1 Summary of modified LeNet model

Figure 1 represents the summary of modified LeNet model. It takes an image of 32×32 size with 1 channel (Grayscale) as an input. The imposition of optimization procedure is carried by Adam and for loss function, Categorical Cross entropy is used. It took 50 ms to run each epoch NVidia GT920M graphics card. Training Accuracy of this architecture was >92% after running 10 epochs. Figure 2 represents the summary of modified VGGNet model consisting of only 8 layers. It takes input as an image of 50×50 size with 3 channels (R, G, B). It took 100 ms to run each epoch NVidia GT920M graphics card. Training Accuracy of this architecture was >95% after running 20 epochs.

We also compare all the three models described and implemented above in order to show the performance of each model. A detection method must anticipate at least 30 frames per second, or 0.033 s per image, to be deemed real-time. The initial CNN model predicts at a rate of 0.072 s per image on average in ten trials, which is $2\times$ slower than the minimum rate required for real-time. Table 3 shows a comparison with various approaches. We can see that the improved LeNet takes 0.030 s per image on average, which is still faster than the simple model. Modified VGGNet outperforms both the models in terms of both the time to predict single image and training accuracy showing accuracy more than 95% which is 3% more accurate than the LeNet while using RGB images as input.

```
Layer (type)                   Output Shape             Param #
=================================================================
conv2d_1 (Conv2D)              (None, 32, 50, 50)          896
_____
conv2d_2 (Conv2D)              (None, 32, 50, 50)         9248
_____
max_pooling2d_1 (MaxPooling2   (None, 32, 25, 25)            0
_____
conv2d_3 (Conv2D)              (None, 64, 25, 25)        18496
_____
conv2d_4 (Conv2D)              (None, 64, 25, 25)        36928
_____
max_pooling2d_2 (MaxPooling2   (None, 64, 12, 12)            0
_____
conv2d_5 (Conv2D)              (None, 128, 12, 12)       73856
_____
conv2d_6 (Conv2D)              (None, 128, 12, 12)      147584
_____
max_pooling2d_3 (MaxPooling2   (None, 128, 6, 6)             0
_____
flatten_1 (Flatten)            (None, 4608)                  0
_____
dense_1 (Dense)                (None, 128)              589952
_____
dropout_1 (Dropout)            (None, 128)                   0
_____
dense_2 (Dense)                (None, 43)                 5547
=================================================================
Total params: 882,507
Trainable params: 882,507
Non-trainable params: 0
```

Fig. 2 Summary of modified VGGNet model

Table 3 Comparison between the models

S. No.	Model	Time to predict single image (ms)	Training accuracy (%)
1	Simple CNN	70	>5
2	Modified LeNet	30	>92
3	Modified VGGNet	20	>95

4.1 Qualitative Findings

To further understand our model's performance, we investigate both successful and unsuccessful examples of detections for qualitative results. A box's confidence is simply the maximum score attained by a box.

Figures 3, 4 and 5 represents the boxes for which the confidence threshold is less than or equal to the confidence and thus such boxes are transposed to Non-Maximum Suppression with the value of 0.1, 0.8 and 0.4. Figure 6 represents the detection of traffic signal and a man with its confidence value. Through experiments we were able to classify most of the objects correctly.

Fig. 3 Output when NMS = 0.1
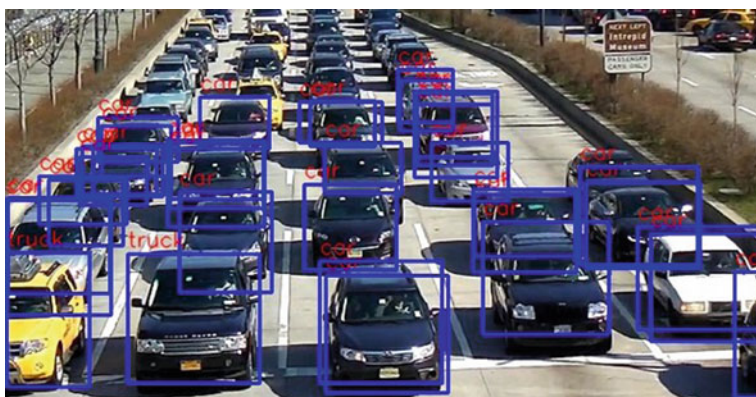
Fig. 4 Output when NMS = 0.8
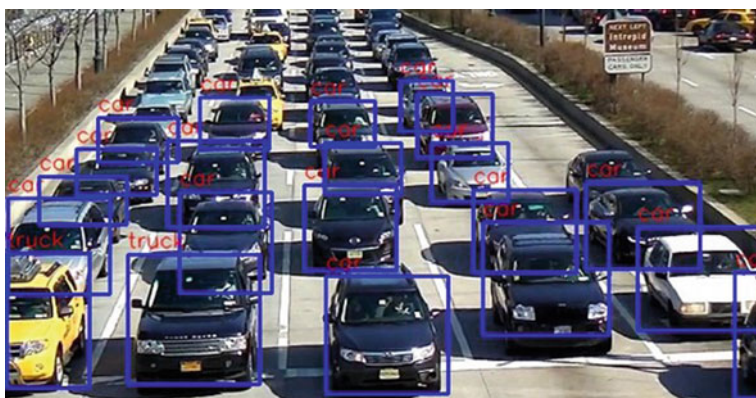
Fig. 5 Output when NMS = 0.4

Fig. 6 Output on traffic signal

5   Conclusions and Future Scope

In this paper, we have put forward the proposal for hybrid CNN deep learning model for real-time identification and classification of objects that makes an image's bounding box predictions without even need for expensive preprocessing or expensive deep evaluations. Three architectures have been defined and used for comparisons including simple, modified LeNet and VGGNet consisting of 8 layers. Along with maintaining high average precision, the model of YONO also facilitate real-time speed and end-to-end training. To make the proposed model more efficient, numerous experiments based upon simulations are conducted. The simulation-based results proved that the modified version VGGNet for CNN model performed very good with RGB images showing an accuracy more than 95% which is 3% more accurate than the LeNet while using RGB images as input.

References

1. Sermanet P, Eigen D, Zhang X, Mathieu M, Fergus R, LeCun Y (2013) Overfeat: integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:182.6229
2. Google's autonomous vehicle. Available: http://googleblog.blogspot.com/2010/10/what-were-driving-at.html
3. Gao HB, Zhang XY, Zhang TL, Liu YC, Li DY (2016) Research of intelligent vehicle variable granularity evaluation based on cloud model. Acta Electron Sin 44(2):365–374
4. Figueiredo L, Jesus I, Machado JT et al (2001) Towards the development of intelligent transportation systems. Intell Transp Syst 88:1206–1211
5. Darpa grand challenge. Available: http://www.darpa.mil/default.aspx
6. Lv C, Wang H, Cao D (2017) High-precision hydraulic pressure control based on linear pressure-drop modulation in valve critical equilibrium state. IEEE Trans Ind Electron
7. Hu X, Wang H, Tang X (2017) Cyber-physical control for energy-saving vehicle following with connectivity. IEEE Trans Ind Electron PP(99):1
8. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. CoRR, abs/1409.1556
9. Redmon J, Divvala S, Girshick R, Farhadi A (2015) You only look once: unified, real-time object detection. arXiv preprint arXiv:1506.02640
10. Eitel A, Springenberg JT, Spinello L, Riedmiller M, Burgard W (2015) Multimodal deep learning for robust RGB-D object recognition. In: 2015 IEEE/RSJ international conference on intelligent robots and systems (IROS). IEEE, pp 681–687
11. Eureka prometheus project. Available: http://www.eurekanetwork.org/project/id/45
12. Silberman N, Hoiem D, Kohli P, Fergus R (2012) Indoor segmentation and support inference from RGB-D images. In: European conference on computer vision. Springer, pp 746–760

Genetic Algorithm-Based Energy-Efficient Clustering with Adaptive Grey Wolf Optimization-Based Multipath Routing in Wireless Sensor Network to Increase Network Life Time

Binaya Kumar Patra, Sarojananda Mishra, and Sanjay Kumar Patra

Abstract The minimum energy consumption issues has now an emerging area of research in wireless sensor network (WSN). These WSN enable new technological application like Internet of things (IoT). In this regard, many researchers delivered various optimization techniques for energy efficiency in WSN. In a wireless sensor network (WSN), the two most important optimization problems are energy-efficient clustering and routing techniques. These two are important to attain the energy efficiency and to increase the network lifetime in a wireless sensor network. Clustering improves scalability and multipath routing by doing optimization makes the WSN more reliable in nature. These two, i.e. clustering and routing, are considered as NP-Hard problem. To handle this type of problem, meta-heuristic optimization is very suitable. In this study, genetic algorithm (GA) is applied in WSNs clustering, and for efficient shortest path communication, ad hoc on demand multipath distance vector routing with adaptive grey wolf optimization (AOMDV_AGWO) is used. The adaptive development of grey wolf optimization technique foresees the optimal path. Therefore, by using AGWO, a near optimal multipath route is chosen from the number of path that is created from ad hoc on demand multipath distance vector (AOMDV) routing protocol. The suggested model undergoes comparison with two existing routing algorithm, i.e. AODV and AOMDV. The simulation outcomes reveals that the proposed Protocol AOMDV_AGWO is more energy efficient as compare to AODV and AOMDV protocol.

Keywords WSN · Clustering · GA · Multipath · AODV · AOMDV · GWO

B. K. Patra (✉) · S. Mishra · S. K. Patra
Department of Computer Science Engineering and Application, Indira Gandhi Institute of Technology, Sarang, India
e-mail: binaya.patra@gmail.com

1 Introduction

Wireless sensor network (WSN) is a mesh of sensor nodes controlled in an ad hoc network. It interacts with the physical world through sensing. Due to its tiny size, a sensor node faces various challenges in energy resources, computing capacity and communication. Nowadays, the continuous development in the area of Internet of things (IoT) by the help of wireless communication along with smart and tiny sensor plays a key role. The most important challenges and issues in design a WSN are the energy consumption. Clustering technique is used to control the data traffic and helps in minimization of energy to increase the reliability of a WSN optimal multipath routing is used. The routing protocol must be act in such a way that it must be an effective and efficient to reduce energy at the time of data transmissions as more energy consumed on that time. Transmission energy directly dependent to the distance between source and destination.

In recent years, many authors suggested different types of routing approach for energy consumption efficiency by using various simulations [1–14]. Different clustering techniques, multipath routing with various bio-inspired optimization in recent days is the efficient techniques that is adopted by most of the researchers. In clustering techniques, whole network is partitioned into different groups of sensors called cluster. In a cluster, a powerful and efficient node is considered as cluster head (CH). The CH has the responsibility for data fusion and transmission. In general, all the source nodes transmit data to CH and cluster head sends the data either directly to the base station (BS) or through other cluster head until it reaches to the destinations node. The reason to use clustering technique is to consume less energy and overcome the hotspot problem. Some of the added parameters that are most relevant to satisfy the efficient implementation of this type of techniques must have to consider are discussed below.

1. Increase network lifetime: Energy conservation directly impacts the network lifetime of a WSN. Proper clustering with perfect CH selection and efficient multipath routing with optimization enhances the network lifetime. From sensor node, hardware architecture to software like clustering and routing protocol must sensitive towards energy efficiency.
2. Data fusion: This happens both at CH and BS, if it is an approach is a clustering one.
3. Latency: In a WSN, data mostly depends on time. Periodic data comes at a certain time interval, and real-time data is updated in each fraction of time.
4. Load balancing: Selection of proper CH and multipath routing protocol balances the network load.
5. Scalability: In a large WSN area to avoid communication overhead, clustering-based technique is best suit with multipath routing. Again the optimization techniques help for scalability.

2   Work Motivations

In a WSN network, practically it is a difficult task to implement optimization techniques. It takes more time to give the accurate output due to upcoming new technological application and requirement because sensor nodes are limited energy, low transmission quality, communication within a short range. These different types of challenges and constraints in a wireless sensor network reduce the performance of the whole system. Whether it is a theoretical approach or a real-time techniques, simulation tools act as a good trade-off to measure the performance like efficiency and accuracy of the system.

In current scenarios, emerging technologies due to their scalability and nonlinearity are complex in nature. To get the improved optimal solution and control the performance for the system modelling, the system requires optimization techniques. Meta-heuristic optimizations are best suit for this type of modelling. For this types of optimization approach, developers needs detail knowledge information of the proposed model.

3   Related Works

In many WSN-related approach, various issues like lower QoS and lower resources are occurring. Due to this, the performance of the whole system reduces. These issues can be resolved by the help of multipath routing techniques. Hossein Jadidoleslamy [1] authored a hierarchical with multipath-based approach called HMR-WSN. In this method, the multipath round is considered as super rounds. Time is divided in to a no of rounds. In every round, CH is selected, and it is informed to the cluster members.

For solving any types of computation, the GA bio-inspired algorithm is used as it is very suitable for resulting in near optimal solution. A number of different methodologies have been suggested in [2–5] how GA-based clustering is efficient for wireless sensor network. Its load balancing nature on sensor nodes enhances the network lifetime.

In [6], GA is used to enhance next-door neighbour node length on the cheap energy usage. Here, distance between sensor nodes to the CH and cluster head to the base station (BS) is regarded as objective function. The node having power this is certainly residual is considered as CH helping to lessen the transmission length between CH to BS. The GA reduces the path length and enhance network lifetime therefore [7]. An improved form of GA is introduced by removing invalid node, hence increase the overall performance of the system. Hierarchical clustering with bio-empowered crossbreed optimization are made by various algorithm strategy this is certainly clustering.

In the proposed approach, [8] described on GA based on the calculation of fitness function by taking parameters like node distance, number of hop count and remaining

energy. It is really resulted as energy efficient. The drawback for the suggested algorithm execution process is slow.

Mekonnen and Rao [9] authored a simulated annealing (SA) with particle swarm optimization (PSO)-based clustering protocol in WSN by taking polynomial time. This clustering comprised of two stages, setup phase and steady-state phase. Then, member nodes have time division multiple access (TDMA) slot allotment policy for sending the information to its corresponding CH. Besides, CH gathers the information from its own member's node and sends to the BS.

Gao et al. [10] presented PSO-based energy-efficient cluster-based routing to reduce the vitality use of the WSN. The node with greater power that is residual energy considered as CH during the clustering. After doing the PSO clustering, the procedure for data transmission ended up being involving between nodes, CH and BS. The data gathering, circulation and aggregation of CH ended up being necessary an energy efficiency; this is certainly additional the WSN.

MLHP [11] is a protocol that is crossbreed and categories the entire network into three spatial levels, considering the length form BS. A GWO-based algorithm can be used to choose appropriate CHs by showing a probabilistic methods of fitness function which counts remaining energy and range neighbours into consideration into the amount this is certainly second level. The CH selection into the amount this is certainly third level distributed as well as a routing tree was created to provide information to the BS.

Grey wolf optimizer (GWO) [12] is just one of the present swarm intelligence algorithms. It is encouraged by the activities of grey wolves, when victim this is certainly hunting. Compared to various other swam intelligence optimization, this has a lot smaller number of parameters. Moreover, it finds its search in a way that both exploitation and exploration are attained. The algorithm is written by it the capability to better approach of local optimum which leads to enhanced convergence [13]. In [14], for multipath routing protocol optimization, an effective Tunicate swarm grey wolf optimization (TSGWO) is suggested. In this, the routing path calculated from taking fitness function as QoS and trust factor. The QoS parameters are energy, delay, distance and link lifetime. In [15], the author proposed a modified grey wolf optimizer for routing protocol for heterogeneous WSN (HMGWO). In this after initial cluster formation, each nodes fitness value is evaluated and considered as the initial weights of GWO. The weights are updated by considering the distance between the wolves. Further to improve the grey wolf optimization, coefficient vector is considered. Some amount of energy, i.e. consumed by all nodes at the time of position change and energy to BS is not considered.

In [16], authors suggested a clustering; this is certainly a new protocol, which utilizes GWO to pick ideal CHs. Clustering is done at the BS, which means protocol that is presented centralized. A relay node is selected for every single of these in order to avoid fast power exhaustion of CHs being not even close to BS. Numerous clustering protocols separate each round into two stages: (a) cluster setup stage and (b) steady-state stage. The clusters formation takes place in the setup phase, and in the steady-state phase, data transmission occurs via CHs. Forming groups presents a power overhead within the community as it results in usage of a quantity that's

sure of as a result of trade of required control packets [17]. This protocol reduces the overhead by removing the setup stage in certain rounds where in fact the CHs from earlier rounds are nevertheless good This should continue till at least one of the CHs has actually consume 50% of their power because the setup that is stand until the setup stage is performed once another time in the starting of the next iteration to find new CHs. If not, the community will carry on to function without doing the setup phase.

In [18], the author proposed a centralized system architecture for IoT application, i.e. C-RAN based C-IoT. The proposed technique is more spectrally efficient and also energy efficient. In [19], the authors evaluated performance analysis RPL protocol. The protocol 6LoWPAN is used for smart home application. This low power and short range protocol used for small IoT application. The RPL performs much better in this environment as against other protocol.

AOMDV may be the most acknowledged multipath on demand routing method, and it is also the advancement regarding the standard AODV protocol. AOMDV locates routes which are numerous the sources towards the destination where it creates the routes that are loop-free and link-disjoint. As compare to AODV, AOMDV protocol prevents the route finding desired in case of route collapse. AOMDV generally considers the hop count as a parameter to find out the path being optimized. The routing table structure of AOMDV [20]. Broadcast hop count can be used as opposed to the usually made use of hop_count in AODV. A routing table list finds the next_hop in case any failure; this modification is considerably identifying next_hops being multiple equivalent hop matters. All next_hops have the same receiver sequence number. In each round, there is a refreshment of sequence number and the initialization of the hop count occurs. AOMDV searches the path which are numerous, send the packet because of the minimum hop count without taking account of energy and traffic in that route, this is certainly which can be seen as the main downside of this model.

4 Radio Energy Model

The radio energy of first order radio is referred from [21] which is shown in Fig. 1. The total energy consumed at transmission time between a transmitter and receiver to send bits over a distance d is given below

$$E_{\text{TX}}(l, d) = E_{\text{tx - elec}} + E_{\text{TX - amp}}(l, d) \tag{1}$$

$$= \{l \times E_{\text{elec}} + l \times \varepsilon_{\text{fs}} \times d^2 \text{ if } d < d0; \tag{2}$$

$$= \{l \times E_{\text{elec}} + l \times \varepsilon_{\text{mp}} \times d^4 \text{ if } d \geq d0; \tag{3}$$
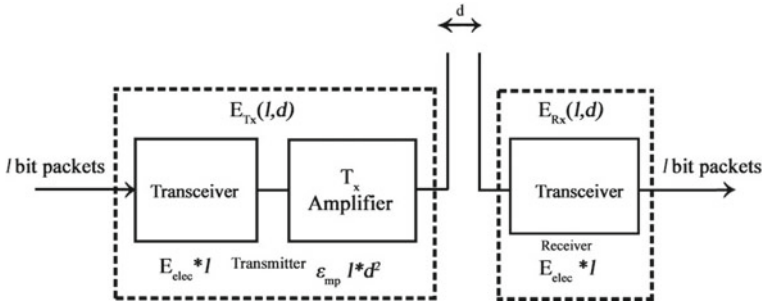
Fig. 1 Radio model

- E_{elec} denotes the dissipated energy per bit. It mostly considers the parameters like signal encoding, digital modulation, filtering and transmission of the signal.
- ε_{fs} and ε_{mp} are the characteristics of the transmitter amplifier, where ε_{fs} is destined for propagation of free space and ε_{mp} meant for multipath.

If $d_0 > d$ (distance between transmitter and receiver), then free space model (d^2 power loss) is implemented.

Else (multipath fading model (d^4 power loss) is used).

The amplifier takes responsibility to reverse this loss. To receive l bit of data, the total energy used and the threshold value is given in (4) and (5), respectively

$$E_{Rx}(l) = \text{Eelec} \times l \tag{4}$$

$$d_0 = \sqrt{\varepsilon_{fs}/\varepsilon_{mp}} \tag{5}$$

5 Projected Methodologies for Clustering

5.1 Genetic Algorithm (GA)

Genetic algorithm is a bio-inspired computational optimization model. For simulation process, it follows biological process and selection. This helps in search for near optimal solution. In this methods, first it identifies the possible solution from current pool of population. Each individual is encoded to act as a characteristic entity for a chromosome. The basic principle GA follows is the survival for the fittest. Once the initial population generated, genetic operator is used to carry out the crossover operation. For new solution set, the mutation operator being carried out. The new population set is now more suitable possible solution as against the previous population set. The fitness value of this offspring is more preferable as previous offspring.

The crossover operator and mutation operator have some specific probability factor. For this reason, it can be efficiently used to get optimal solution in any computational model.

5.2 Procedure for GA Clustering

/* for each individual Iteration */

Implement binary encoding for GA procedure

Follow GA methods for selection of CHs in each round, i.e. $K_{optimal}$

Step-1 Population initializations.

Step-2 Generates the chromosome record by taking binary encoding.

Step-3 Selection of chromosomes randomly.

Step-4 Evaluate the fitness function utilizing (6).

Step-5 Carry out the crossover and mutation phase to produce next generation of population.

Step-6 The new generation of population considered as off springs.

Step-7 For the new population, create the new cluster.

Step-8 New cluster head selected by using fitness value.

Step-9 If any round, the fitness value is same for more than one node, then the node having shortest distance to base station considered as CH.

/*All CHs send the collected information to BS, and then BS gathers all the data, carry out next phase of work.*/

end of the process

/* stop itteration */

5.3 GA Process

The various steps of GA included in this approach are discussed below.

1. Encoding: In this work, binary encoding technique is applied, which categories the entire nodes of the network in to two. One is considered as members nodes and others are considered as cluster head (CH). Member sensor nodes and CH regarded as gene to generate chromosome. The members node and CH taken as (0) and (1) for calculation of chromosome creation.

2. Initializations of population: Initial chromosome created by the input of certain number of nodes, and it can be increased to some number. For the space constraint and easy to execute, the chromosomes length should be restricted.

3. Fitness function calculation: In this work, it is the ratio of residual energy to total energy of the sensor nodes. Mathematically, the equation is defined as below

$$F_1(x) = \frac{E_r}{E_i} \tag{6}$$

where E_r = sensor node residual energy and E_i = sensor node initial energy.

4. Selection process: In this phase from the present population, individuals are chosen for next generation of population. By doing selection process, better fit individuals selected for reproduction of next population. Roulette Wheel selection method is used for chromosome selection method.

5. Crossover operation: In this phase, single point crossover method is used. The crossover between two individual chromosome having probability specified by crossover rate.

6. Mutation: In each bit of chromosome, mutation operator is applied. The mutation rate has some probability specification. After the mutation, each bit changes, i.e. 0 becomes 1 and vice versa. In this work, the chromosome selection, for the evaluation of CH, the fitness value used the probability function

The probability function that is used for selection of CH based on the fitness value is evaluated as:

$$P\left(CH_{\text{chromosome}(i)}\right) = \frac{(Fx_i)}{\sum_{M}^{N} F(x_M)} \tag{7}$$

6 Proposed Ad Hoc on Demand Multipath Distance Vector Routing with Adaptive Grey Wolf Optimization (AOMDV-AGWO) Algorithm

The transmitter broadcasts a path demand (RREQ) to get the path to the receiver node into the AOMDV routing protocol [16] and multiple number of path created. Among all the routes, AOMDV chooses the shortest one, i.e. considering minimum hop count and does not also thinking about the quality of this selected route. The transferred data packet from source to destination, when received message from destination node returns to the source node. In this regard, more than one number of transmissions occurs in different preferred routes. So, it hampers the link quality loss. For the possible loss of connection, source node does not accept any acknowledgement. The system decides one optimal route. Accordingly in this regard, a meta-heuristic optimization methods based on grey wolf optimization are used. Here in this approach,

fitness function is used to make it adaptive in nature. The proposed routing algorithm with optimization technique is called AOMDV_AGWO.

6.1 Adaptive Grey Wolf Optimization (AGWO)

AGWO is influenced because of the food hunting activities of the grey wolf. A wolf is called subsidiary or as delta sometimes if it does not are parts of the blended crowd of alpha (α), beta (β) or omega (δ). The adaptive methods implies after enhance wolf actions and employ Genetic parameters [12].

For resourceful information sending, find the optimality of route from selected number of routes. Into the targeted node during the selected number of routes, the information transmission from the source node in the community is done by the route request message. The route is delivered by the destination node acknowledged message towards the source node after obtaining the received packet. The remainder energy regarding the nodes must be well-ordered and route among the list of sender node and receiver node ought to be shortest to acquire the suitable path.

A number of solutions (S_a) in the search space area are started initially. For route optimality, node must consume less energy, and there must be a shortest optimal path exist between sender to receiver. A fitness function for the suggested algorithm is given below:

$$\text{Fitness Fuction} = \min\left\{\sum_{i=1}^{s} \text{Distance}(n, m_{i+1})\right\} \tag{8}$$

6.1.1 Updating of AGWO Process

1. For different route initialization of grey wolf process.
2. Evaluate the fitness function of each route.
3. Categories the route based on the fitness value.
4. Here. P_α, P_β, $P\delta_{\text{represents}}$ the first, second and third best route, respectively.
5. Upgrade the position vector of the present explores solutions.

$$P(t + 1) = (P_1 + P_2 + P_3)/3$$

6. Update the a, A and C parameter.
7. Calculate the location information of the each wolves individually.
8. Crossover and mutation process again updated.
9. The process should continue until an optimal route solution comes out.

6.1.2 Evaluation of Position Vector

Let us assume for the grey wolves that the wolf alpha considered as top contender for result. That is well along with beta and delta are best suited because of the awareness; this is certainly superior the likely region of the prey because of the purpose of hunting behaviour [18].

$$P_1 = P_\alpha - A_1.(R_\alpha), P_2 = H_\beta - A_2.(R_\beta), P_3 = H_\delta - A_3.(R_\delta) \tag{9}$$

$$R^\alpha = |C_1.P_\alpha - P|, R^\beta = |C_1.P_\beta - P|, R^\delta = |C_1.P_\delta - P| \tag{10}$$

A comment that is last the GWO is the preserving associated with the different parameters that controls the trade-off among the exploration and exploitation.

$$\left.\begin{array}{l} A = 2a.r_1 - a \\ C = 2.r_2 \end{array}\right\} \tag{11}$$

The major alteration parameters for GWO are "*a*" and "*C*". For adaptation, the two main parameters are "*A*" and "*C*". In this approach to make the GWO in to simpler form, minimum numbers of operators to be considered.

6.1.3 Adaptive Process

In this approach, single point crossover operator is applied for optimal solution. The condition applied for termination is small crossover rate with greater mutation rate.

6.1.4 Crossover and Mutation Process

In this phase, the total population is divided in to two parts: one is crossover, i.e. exploitation and other is the adaptive mutation, i.e. exploration. Population diversity plays a major role for evaluation of these division sizes.

7 Performance Evaluation

7.1 *Simulation Settings*

To check the performance of the model, various parameters are taken for consideration. The MATLAB simulation tool is used. In Table 1, the experiments are carried

Table 1 Simulation parameters

Parameter	Value
Routing protocol	AOMDV, AODV, AOMDV-AGWO
Simulation tool	MATLAB
Node coverage area (x, y) in metre	500 m, 500 m
Sensor nodes (n)	100 Nos
Initial energy in joules	100
MAC Type	802.11
Crossover rate (clustering)	0.5
Mutation rate (clustering)	0.001
Simulation time in second	100
Node mobility in metre/second	10 m/s
Traffic type	CBR
Propagation	Free space
Antenna	Omni
Mobility	Random way

out by taking 100 nos of nodes and are distributed randomly in a 500 m × 500 m WSN area.

7.2 Estimation Metrics

Some suitable metrics are selected to measure the performance of the suggested protocol. These are described as below:

- **Throughput**: It is calculated as the total no. of bits effectively delivered at the receiver node in the WSN. In Fig. 2, the performance of the throughput has been

Fig. 2 Throughput versus no. of nodes
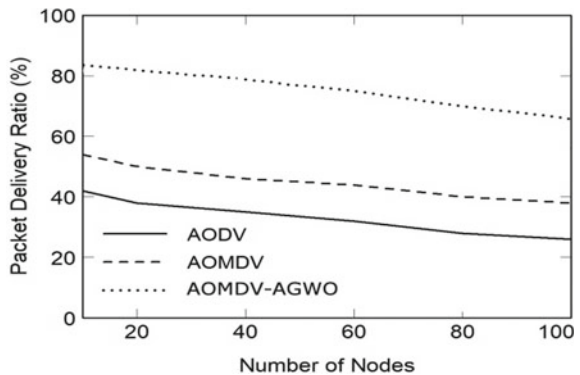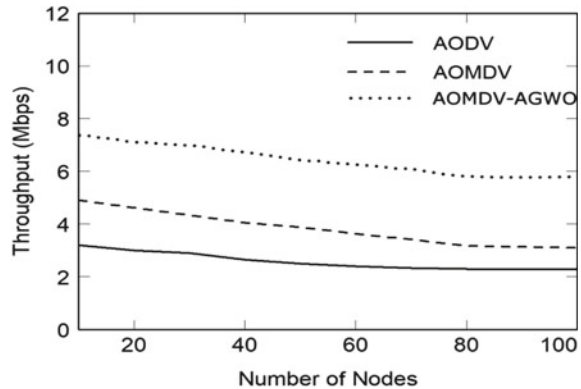
shown. In this AODV, AOMDV and the proposed protocol (AOMDV-AGWO)
have been analysed. The proposed protocol showing better performance than
other two. It sends the data packet in the best route considering highest level of
energy with shortest source to destination distance.

- **Packet Delivery Ratio (PDR)**: This is a ratio of number of packets received by
  the receiver node to the total packets source node sends. In Fig. 3, the mobility
  speed of PDR effect according to the figure shown is 10 m/s. If number of nodes
  increases in the networks, it increases the traffic, and hence, there is possibility of
  congestion. For this reason, packet drop increases and PDR reduces. By using the
  fitness function, the PDR of the proposed protocol is best as against other two.
- **Routing Overhead Ratio (ROR)**: For the purpose of route discovery and route
  maintenance, the number of packet requirement for broadcasting the delivery of
  packets. The number of routing over head packets depends on different parameter
  for better performance. These parameters are mostly stability of the network,
  routing protocol, size of the network and the topology of the network. AODV
  creates one best route and AOMDV creates multiple number of routes. For this
  reason, AOMDV requires more routing packets. So AODV has less ROR. In
  this proposed model, as fitness function is used so the routing overhead packets
  increases. Hence in Fig. 4, it is understand that the proposed model requires more
  overhead than other two.
- **Energy Consumption**: During the simulation time, the total energy consumed
  by each nodes in WSN. The main aim of our protocol is to consume less energy.
  In Fig. 5, it is clearly showing that the proposed protocol is more energy efficient
  than the other two.

It is clear that the proposed protocol performs best in case of packet delivery
ratio, energy consumption and throughput. It shows a poor performance in routing
overhead ratio.

Fig. 4 Routing overhead versus no. of nodes
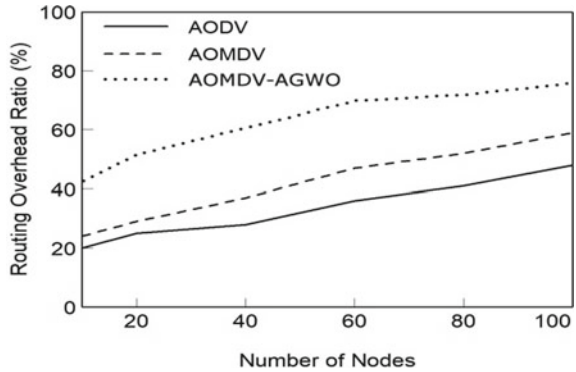
Fig. 5 Energy consumption versus no. of nodes
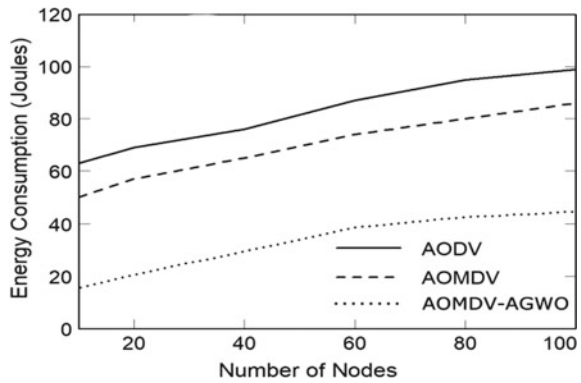
8 Conclusions and Future Work

In various study, it is observed that bio-inspired clustering with meta-heuristic multi-path routing optimization present a scalable routing in a large WSN application. In this approach, GA-based clustering with adaptive GWO protocol is used to get near optimal solution. In the present real-world scenario and ever growing challenges, the proposed model could provide a quick response. The security part of the model is not considered. In future, work security measure will be considered for various types attack like wormhole attack, selfish node attack and malicious node attack.

References

1. Jadidoleslamy H (2017) A hierarchical multipath routing protocol in clustered wireless sensor networks. Wirel Pers Commun 96(3):4217–4236
2. Bari A, Wazed S, Jaekel A et al (2009) A genetic algorithm based approach for energy efficient routing in two tiered sensor networks. Ad Hoc Netw 7:665–676

3. Pan Y, Liu X (2007) Energy-efficient lifetime maximization and sleeping scheduling supporting data fusion and QoS in multi-sensor net. Signal Process 87(12):2949–2964

4. Srinivas N (1995) Multiobjective optimization using nondominated sorting in genetic algorithms. J Evol Comput 2(3):221–248

5. Somauroo A, Bassoo V (2019) Energy efficient genetic algorithm variants of PEGASIS for 3D wireless sensor networks. Appl Comput Inform

6. Aziz L, Raghay S, Aznaoui H, Jamali A (2016) A new approach based on a genetic algorithm and an agent cluster head to optimize energy in wireless sensor networks. In: 2016 İnternational conference on information technology for organizations development (IT4OD), pp 1–5

7. Yao G-S, Dong Z-X, Wen W-M, Ren Q (2016) A routing optimization strategy for wireless sensor networks based on improved genetic algorithm. J Appl Sci Eng Technol 19:221–228

8. Hamidouche R, Aliouat Z, Gueroui AM (2018) Genetic algorithm for improving lifetime and Qos of wireless sensor networks. Wirel Pers Commun 101(4):1–37

9. Mekonnen MT, Rao KN (2017) Cluster optimization based on metaheuristic algorithms in wireless sensor networks. Wireless Pers Commun 97(2):2633–2647

10. Gao F, Luo W, Ma X (2019) Energy constrained clustering routing method based on particle swarm optimization. Cluster Comput 22(3):7629–7635

11. Al-Aboody NA, Al-Raweshidy HS (2016) Grey wolf optimization-based energy-efficient routing protocol for heterogeneous wireless sensor networks. In: 2016 4th international symposium on computational and business intelligence (ISCBI)

12. Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Adv Eng Softw 69:46–61

13. Faris H, Aljarah I, Al-Betar MA, Mirjalili S (2017) Grey wolf optimizer: a review of recent variants and applications. Neural Comput Appl 30(2):413–435

14. Chouhan N, Jain SC (2020) Tunicate swarm Grey Wolf optimization for multi-path routing protocol in IoT assisted WSN network. J Ambient Intell Hum Comput. https://doi.org/10.1007/s12652-020-02657-w

15. Zhao X, Ren S, Quan H, Gao Q (2020) Routing protocol for heterogeneous wireless sensor networks based on a modified grey wolf optimizer. https://doi.org/10.3390/s20030820

16. Mahdi SM, Daneshvar H, Mohajer PAA, Mazinani SM (2019) Energy-efficient routing in WSN: a centralized cluster-based approach via grey wolf optimizer. IEEE Access https://doi.org/10.1109/ACCESS.2019.2955993

17. Hosen A, Cho G (2018) An energy centric cluster-based routing protocol for wireless sensor networks. Sensors 18(5):1520

18. Mahapatra B, Turuk AK, Patra S.K (2020) Exploring power consumption reduction in centralized radio access for energy-efficient centralized-Internet of things implementation. https://doi.org/10.1002/ett.4045

19. Sanila A, Mahapatra B, Turuk AK (2020) Performance evaluation of RPL protocol in a 6LoWPAN based smart home environment. IEEE Xplore

20. Marina MK, Das SR (2001) On-demand multipath distance vector routing in ad hoc networks. In: Proceedings of 9th ınternational conference on network protocols (ICNP), pp 14–23

21. Alkesh A, Singh AK, Purohit N (2011) A moving base station strategy using fuzzy logic for life time enhancement in wireless sensor network. In: Proceedings of ınternational conference on communication systems and network technologies, Katra, Jammu, India, pp 198–202

Power Control of a Variable Speed Wind Turbine Using RBF Neural Network Controller

Satyabrata Sahoo

Abstract In this paper, fuzzy logic and radial basis function-oriented (RBF) neural network controller is implemented and compared for a 2 MW wind turbine which operates in the region-II wind speed. Here, through the use of collective pitch angle, generator power is controlled in region-II wind speed zone. The novelty of this paper is the turbine output power quality that is compared using PI, fuzzy logic, and RBFNN. The wind turbine and all of its accessories are designed using MATLAB/SIMULINK, and the control schemes are implemented to get the desired power output. From the obtained results, it is found that the RBF-oriented NN controller exhibits an excellent response compared to other existing technique to prove its superiority. The work is verified through the results of MATLAB/SIMULINK software.

Keywords Wind turbine · Region-II · Fuzzy logic control · Neural network control · Radial basis function

1 Introduction

Generation of wind electrical power is increasing rapidly due to its less environmental problems and lower cost. Based on world wind energy association, its progress level is approximately 30% [1]. The wind power generated from the wind turbine is dependent on third power of wind speed. Because of this, it is treated as a nonlinear system. In fact, the main goal is to extract the power from the speed of wind. Hence, to achieve this, a major role is being played by the control system. In Fig. 1, power operating region of a variable speed variable pitch (VSVP) wind turbine is presented. Mainly, it consists of two operating regions, i.e., variable speed region and variable pitch region. The variable speed/first region is in between 4 and 12 m/s of the wind speed. Similarly, the variable pitch/second region is in between 12 and 27 m/s of the wind speed. This research manuscript emphasis only on the variable pitch/second

S. Sahoo (✉)
Department of Electrical and Electronics Engineering, Nalla Malla Reddy Engineering College, Hyderabad 500088, India
e-mail: jitu_sahoo@yahoo.com
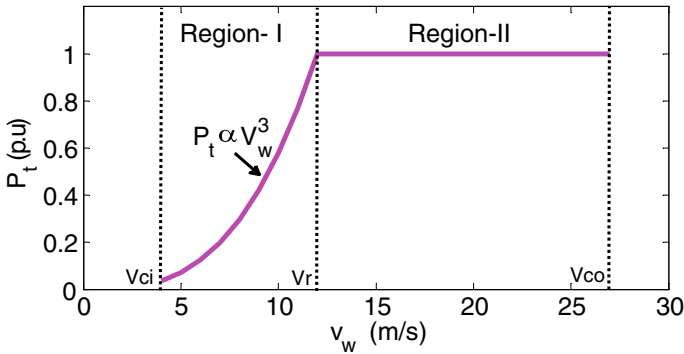
Fig. 1 Power curve characteristics for WECS

region. As the power output exceeds the turbine generator rated value, here, it is regulated through generator torque and blade pitch angle.

The VSVP wind turbine systems are preferred due to their better quality power output, less aerodynamic load, and higher efficiency. In control of wind turbine, classical control technique like proportional-integral and derivative (PID) control is widely exercised by separate researchers in papers [2–4]. Because of low cost and easy application, PID controllers are widely used in industry even today.

It is observed that the conventional control method like PIDs are unsuitable for handling complex or ill-defined systems. Intelligent systems which combine knowledge, techniques, and methodologies from various sources are alternative approaches for resolving the imprecise dynamical behavior of such complex systems. Fuzzy logic is the one such intelligent technique. In fuzzy logic, alternative approaches like decision-making processes, qualitative aspects of human reasoning, and alternative to capture the approximate are created by using fuzzy inference systems. Therefore, by means of fuzzy logic, human languages are expressed. Also, fuzzy rules are created by the use of human expertise and experience. Through fuzzy logic controller, rules are created, and it converts a linguistic control strategy in to an automatic control strategy.

Hence, application of fuzzy logic control in wind energy conversion technology problem through turbine pitch angle is a suitable choice [5–7].

In FLC, the rules are heuristic in nature, and the design method depends on the experience and knowledge of the operator. To tackle this hindrance, an alternative intelligent technique such as artificial neural network (ANN) is utilized in WECS. Here, input and output data are used to model the system, even without knowing about the system. It is because of its learning capability. ANN application in variable speed wind turbine is listed [8–10]. Though a lot of research papers published in ANN, still, there is a lot more scope to improve on it.

In this paper, we introduce a fuzzy logic and neural network controller to control the power of the variable speed and variable pitch wind turbine in the wind speed of region-II. We use the radial basis function type neural network, to get proper tracking

output power from the region-II zone. The rest of the paper is organized is as follows. Section 2 presents the mathematical modeling of wind energy conversion system. Section 3 presents the details of the control algorithm implementation in wind turbine to control the power. Sections 4 and 5 present the results analysis and conclusions of this work, respectively.

2 Modeling of Wind Turbine

The kinetic energy of wind is transformed to mechanical energy through the blades of the wind turbine. Then, this mechanical energy is converted to electrical energy through the electric induction generator. The wind power [11, 12] available at the rotor disk is

$$P_{\text{wind}} = 0.5\rho A v^3 \tag{1}$$

where A is the swept area in m^2, ρ is the air density, v is the wind speed in m/s.

Due to the friction, losses are present in a mechanical rotational system. A fraction of wind power available at the rotor disk is to be converted to rotor turbine power. This fraction is known as the turbine-efficiency power coefficient C_p, and its magnitude is never exceeded to 0.593.

$$C_p = \frac{P_t}{P_{\text{wind}}} \tag{2}$$

where P_t is the power available at the rotor turbine. It is expressed as

$$P_t = P_{\text{wind}} C_p(\lambda, \theta) \tag{3}$$

Here, C_p is a function of tip speed ratio and pitch angle θ. The tip speed ratio is the ratio of blade tip speed to the speed of wind and is expressed as

$$\lambda = \frac{R\omega_r}{v} \tag{4}$$

where ω_r is the rotor speed in rad/s and R is the radius in m.

The mechanical torque T_m in 'Nm' is given by

$$T_m = \frac{P_{\text{wind}}}{\omega_r} = 0.5\rho A R v^3 C_q(\lambda, \theta) \tag{5}$$

where C_q torque coefficient and is expressed as
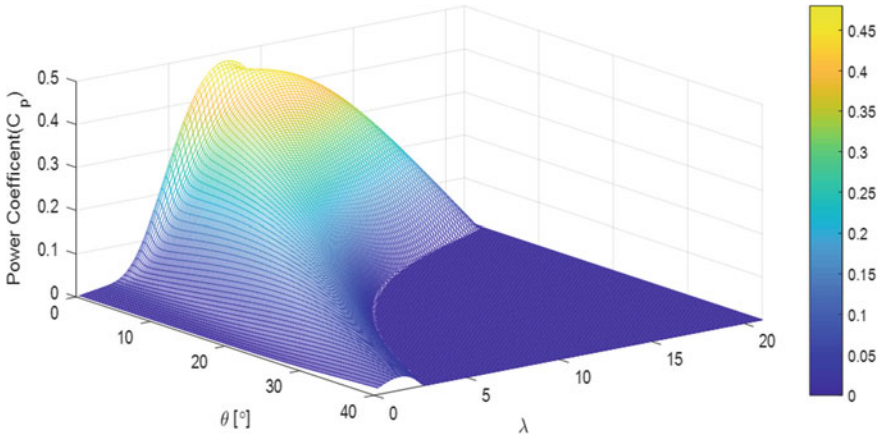
Fig. 2 Torque coefficient with lambda and theta for calculation of mechanical torque

$$C_q(\lambda, \theta) = \frac{C_p(\lambda, \theta)}{\lambda} \tag{6}$$

The electrical power P_e in watt is given by

$$P_e = T_g \omega_g \tag{7}$$

where T_g is the generator torque in N/m and generator rotor speed ω_g in rad/s and $\omega_g = n_g \omega_r$ A torque coefficient curve for different tip speed ratio and pitch angle is shown in Fig. 2.

3 Controller Design

3.1 Fuzzy Logic Controller

Proportional and integral controllers cannot perform well with the control law of variable speed of wind turbine because of linearization problem and nonlinearity in power system. These problems can be eradicated by the use of fuzzy control.

In this paper, a fuzzy logic-based controller is used and shown in Fig. 3. Here, two errors, i.e., *e* and *de* are the two inputs to the fuzzy controller design. These inputs are actually generator power and its speed. The output of the fuzzy controller is generator torque and desired pitch angle. The fuzzy logic membership functions, type of membership function, and rules are implemented here from [6].
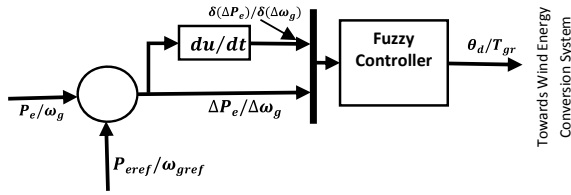
Fig. 3 Fuzzy logic controller used in WECS

3.2 RBF Neural Network Controller

Artificial neural network is a part of machine learning or artificial intelligence. With the availability of suitable database, ANN is able to learn, store, and recall from the database in an appropriate manner [8]. Radial basis function neural network is one of the dominant approaches of the ANN. The structure of a RBFN is shown in Fig. 4, which consists of three layers, i.e., the input layer, hidden layer, and output layer. By doing so, it makes the modeling competent. Each layer comprises of number of neurons, and RBF retains as a hidden layer. Hence, the input vectors are directly processed to the RBF layer, after transformation to a nonlinear activation function through the neuron [10].

With the availability of different functions, radial basis Gaussian transfer function is used here. Mathematically,

$$f(x, b, \tau) = \exp^{\left\{-\left(\frac{x-b}{\tau}\right)^2\right\}} \tag{8}$$
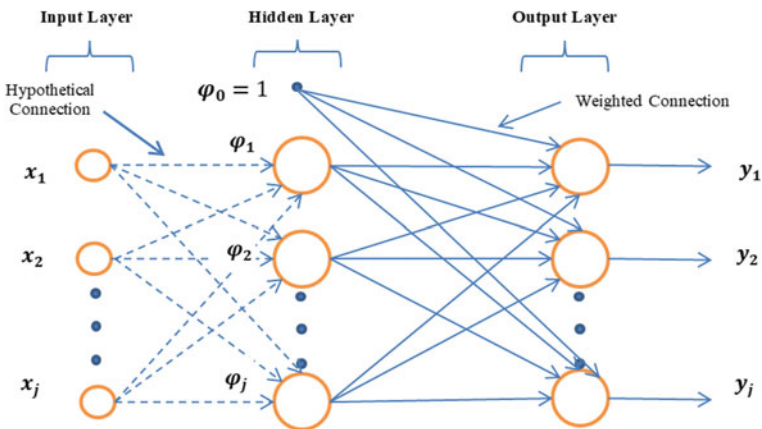
Fig. 4 Typical RBFNN structure

where τ is the variance, x is the input, and b is the center. The output of the network at time t is given by

$$y_j(t) = \sum_{k=1}^{n} \omega_{jk} f_j(x, b, \tau) \tag{9}$$

where ω is the weight of the connection string.

For the development of RBF controller, the simulated data are collected by using conventional controller. The data collected from the conventional controller is optimized by Ziegler–Nichols method. In total, 1000 samples or simulated data are collected. The control block diagram and convergence curve are shown in Figs. 5 and 6, respectively.
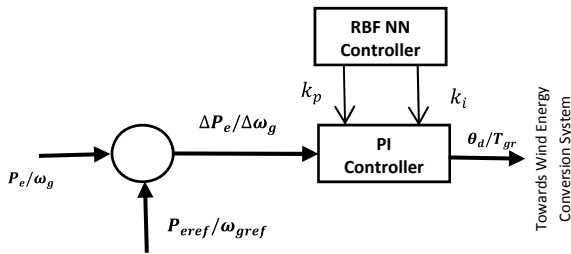
Fig. 5 Bock diagram of RBFNN controller
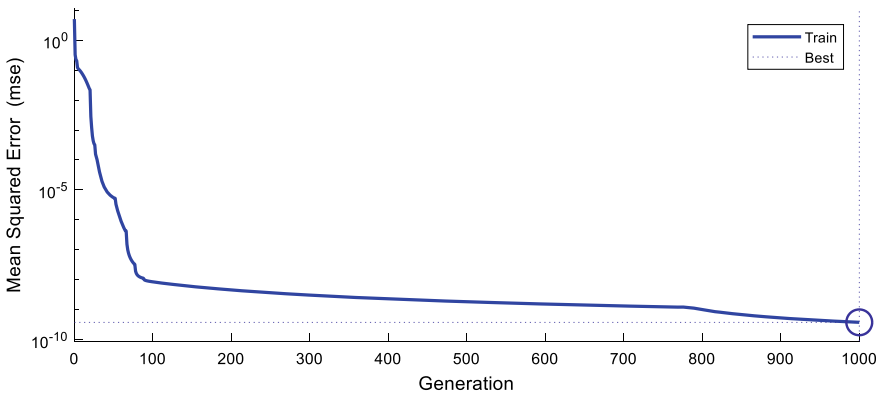
Fig. 6 Convergence diagram of RBFNN
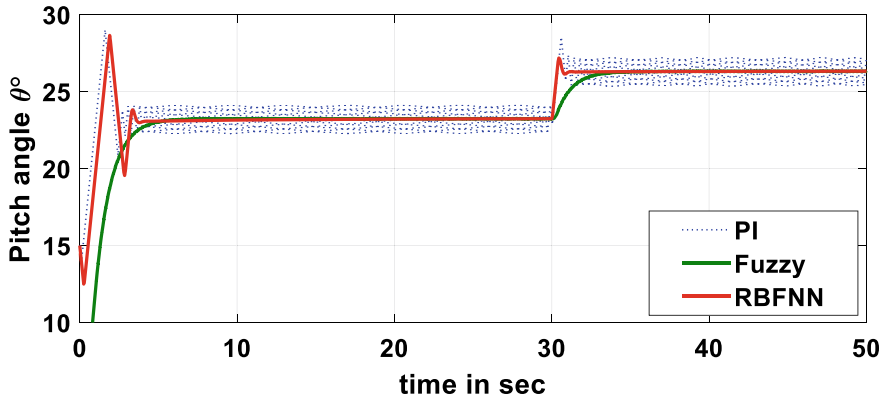
Fig. 7 Pitch angle comparison using up wind speed

4 Results and Analysis

The output results of a variable speed wind energy conversion system with above rated speed of wind are shown in this section. The simulation program for this is also carried out through MATLAB/SIMULINK software with PI, fuzzy logic, and RBF neural network controller.

The aforementioned results of MATLAB are based on two different ranges of wind speed, i.e., up wind and down wind. The step wind speed is used here to make the system more robust. The wind speed changes from 15 m/s to 17 m/s at 30 s out of total 50 s of simulation. From Figs. 7 and 9, it is observed that the pitch angle control is smoother in case of RBFNN as comparison to PI and fuzzy logic method. As the deviation is less, the stress on the pitch actuator motor will be less, and so, the motor life span is increased. Similarly, from Figs. 8 and 10, it is observed that the output power is smoother in case of RBFNN in comparison to other two methods. As the oscillation in magnitude of power generation is reduced, harmonics reduced, and finally, the quality of power is improved. Because of improved power quality, the magnitude of power generation is also increased. Due to this, the projected power tariff is also to be reduced for the power consumers.

5 Conclusions and Future Scope

In this paper, the quality of output power from the wind turbine model is studied by using PI, fuzzy logic, and RBFNN controller. The quality of power in terms of fluctuation is measured here. Along with the conventional PI and fuzzy logic controller, the ANN is used here because of its better learning, storing, and recalling capacity from the database. The fluctuations of power are higher for the case of generator power and pitch angle output when used the conventional PI controller. These deviations
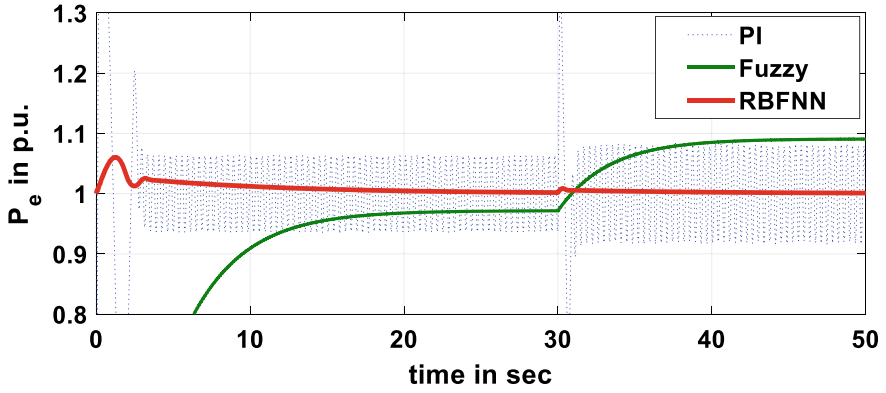
Fig. 8 Output power comparison using up wind speed
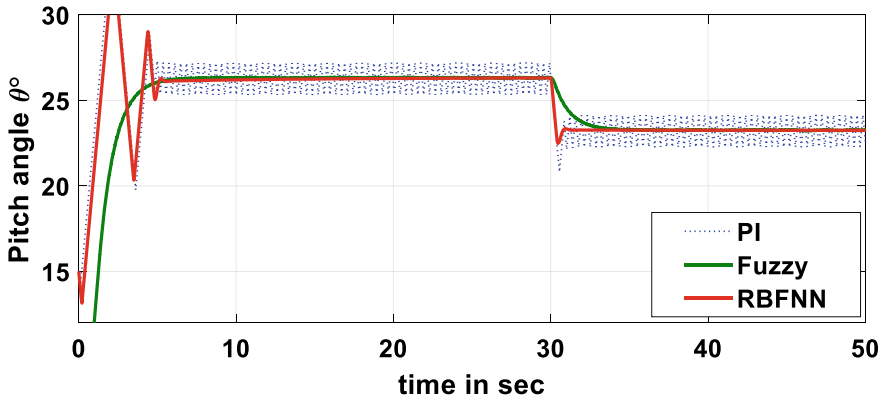
Fig. 9 Pitch angle comparison using down wind speed
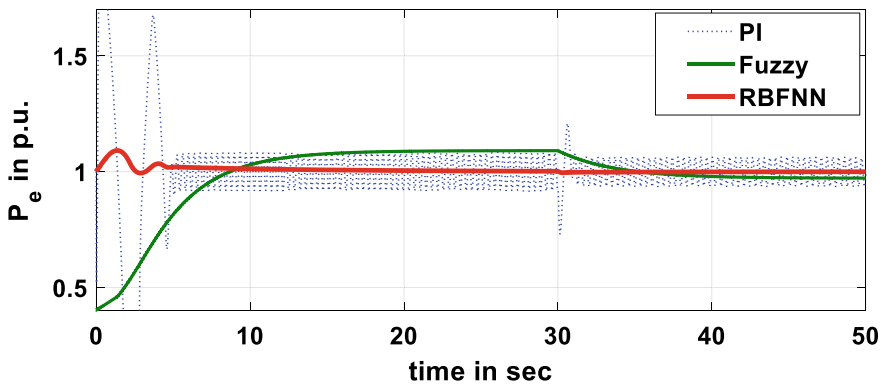
Fig. 10 Output power comparison using down wind speed

are reduced for RBFNN and fuzzy logic case. But, the result produced by RBFNN is the best one.

As the output power fluctuations are reduced, total harmonic distortion is reduced. So, also, the power quality is improved. Again, as the pitch angle oscillation is decreased, the longevity of the actuator motor is also enhanced for the wind turbine.

Hence, among PID, fuzzy, and RBFNN controller, the RBFNN controller is verified to be more applicable and achievable in regulating the power output of a variable speed wind turbine through control of pitch angle. In future scope, this model is to be implemented real-time application, and the results are to be verified through LabVIEW.

References

1. Boukhezzara B, Lupu L, Siguerdidjane H, Hand M (2007) Multivariable control strategy for variable-speed variable pitch wind turbines. Renew Energy 32(8):1273–1287
2. Apata O, Oyedokun DTO (2020) An overview of control techniques for wind turbine systems. Sci Afr 10.
3. Benamor A, Benchouia MT, Srairi K, Benbouzid MEH (2019) A new rooted tree optimization algorithm for indirect power control of wind turbine based on a doubly-fed induction generator. ISA Trans 88:296–306
4. Pan LH, Wang X (2020) Variable pitch control on direct-driven PMSG for offshore wind turbine using Repetitive-TS fuzzy PID control. Renewable Energy 159:221–237
5. Zamzoum O, Mourabit YE, Errouha M, Derouich A (2018) Power control of variable speed wind turbine based on doubly fed induction generator using indirect field- oriented control with fuzzy logic controllers for performance optimization. Energy Sci Eng 6:408–423
6. Sahoo S, Subudhi B, Panda G (2016) Pitch angle control for variable speed wind turbine using fuzzy logic. In: IEEE international conference on information technology (ICIT), IIIT, Bhubaneswar, India, pp 28–32
7. Civelek Z, Lüy M, Çam E, Mamur H (2017) A new fuzzy logic proportional controller approach applied to individual pitch angle for wind turbine load mitigation. Renew Energy 111:708–717
8. Jie W, Jingchun C, Lin Y, Wenliang W, Jian D (2020) Pitch control of wind turbine based on deep neural network. IOP Conf Ser: Earth Environ Sci 619:1–8
9. Yilmaz AS, Özer Z (2009) Pitch angle control in wind turbines above the rated wind speed by multi-layer perceptron and radial basis function neural networks. Expert Syst Appl 36:9767–9775
10. Barbade SA, Kasliwal P (2012) Neural network based control of doubly fed induction generator in wind power generation. Int J Adv Res Technol 1(2):1–5
11. Bianchi FD, Battista H, Mantz RJ (2007) Wind turbine control systems: principles, modelling and gain scheduling, 1st edn. Springer, London
12. Soliman M, Mallick OP, Westwick D (2011) Multiple model MIMO predictive control for variable speed variable pitch wind energy conversion systems. IET Renew Power Gener 5(2):124–136

Machine Learning Approach for Change Detection of Chandaka Wildlife Sanctuary with the Help of Remote Sensing Data

Arpita Majhi, Kabir Mohan Sethy, and Mrutyunjaya Panda

Abstract Machine learning is a learning process where computer data analysis can be modeled without any human interference and carry out certain tasks. Change detection of forest cover is very vital for the environment and its applications. It is a method to detect how the area of a particular region changed over different periods. It can be caused due to various reasons either by natural or manmade events. It is to evaluate urban growth, deforestation, and other natural calamities. Remote sensing is a process of collecting field data and other information about an area from a distance with the help of a satellite. Chandaka Forest is a wildlife sanctuary since August 1982. It covers an area of 193.39 km^2 and is situated in the Khurdha District adjacent to Bhubaneswar, the capital city of Odisha, India. This paper studied a part of Chandaka Forest for change detection of forest cover from the year 2004 to 2020 using image processing in MATLAB Software. Image processing is a process to study an image to get useful information. The methods applied in this study are image preprocessing, Change Detection Algorithm, Threshold-Based image segmentation. The finding is an increase in deforestation from 2018 to 2020, but in the year 2020 changes increased drastically.

Keywords Remote sensing · Image processing · Forest cover · Change detection · MATLAB

1 Introduction

Chandaka Forest is a wildlife sanctuary since August 1982. It covers an area of 193.39 km^2 and is situated in the Khurdha District adjacent to Bhubaneswar, the capital city of Odisha, India. It lies in the Northwest of Bhubaneswar city; it is an eco-sensitive zone. It contains 11 demarcated protected forest blocks, 7 reserved

A. Majhi (✉) · M. Panda
Department of Computer Science and Application, Utkal University, Bhubaneswar, Odisha, India
e-mail: arpitaexcel123@gmail.com

K. M. Sethy
Department of Geography, Utkal University, Bhubaneswar, Odisha, India

forest blocks, protected forests, private lands, forest lands, government wastelands under the revenue department. It is located between 850 34′ 42″ E–850 49′ 27″ E longitude and 200 12′ 29″ N–200 26′ 18″ N latitudes and is encompassed in Survey of India toposheet No.73H/11, 12, and 15 [1]. In the year 2006, the area statistics of forest land use classes of the sanctuary are obtained using digital image processing technique and geographic Information System [2].

Remote Sensing Technology is important in collecting data for productive resource direction and helpful in monitoring the environment. It is a process of collecting field data and other information about an area from a distance with the help of a satellite. The satellite data helps the researchers create an interest for detecting changes with that image of the same location at different times. For the periodic tracking and analysis of changes in period, remote sensing data are proven best for data collection. Various applications are present to detect the changes in the images.

Nowadays the improvement in remote sensing technology has increased and high-resolution satellite images are easily available. Previously the field data collection was very time-consuming and difficult, now it can be easily accessible by the researchers. Today satellite data made data collection easier and less time-consuming, with the help of these data one can accurately detect the changes in the environment. The motivation in understanding the change detection is to understand the environment better and the role of human interaction. There are steps like data collection, image preprocessing, segmentation, classification, and change detection algorithms. The significance of the study is to compare the deforested area in a different period using remote sensing data, i.e., satellite images. This paper contains an outline of the change detection framework, data collection, image preprocessing technique, change detection method, image segmentation, accuracy, result and discussion, conclusion and future scope. The aim of the study is to detect the changes in the forest in different period.

2   Change Detection Framework

Changed detection framework is designed to identify remarkable changes in different images in different time periods [3]. The changes detected constitute a change map. The methodologies in the change detection framework are shown in Fig. 1. The process starts with the collection of images of the same location obtained at different time periods. The image preprocessing helps in filtration. The change detection is identified where image segmentation will determine the changes in area or no changes in the area.
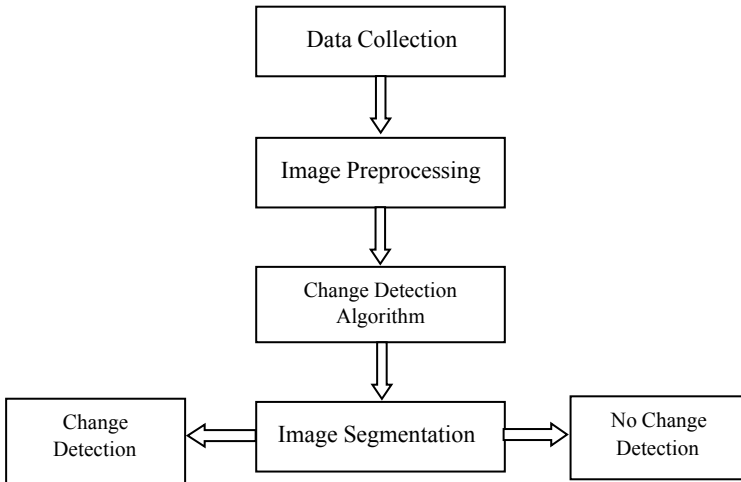
Fig. 1 Change detection framework

2.1 Data Collection

The data collection, i.e., remote sensing image collection of an area for different years is challenging. The images acquired from the remote sensors are actually diverse in terms of intensity, color, and wavelength. The information of the image is observed in terms of pixels. The type sensor determines the resolution of the image obtained from the satellite. Satellite images are easily available and collected from satellite sensors like Landsat, Quickbird, and Spot satellites. For the change detection most commonly and frequently used remote sensors are Landsat imagery.

2.2 Image Preprocessing

Image preprocessing is a method that helps in removing unwanted noises like the atmospheric inferences of the images. There are various procedures to remove the noises from the remote sensing images. Researchers find difficulty in differentiating the change detection with the human eye. Choosing a preprocessing technique is important for good change detection which increases the accuracy and efficiency. Radiometric correction caused by the source light and angle of the sensor while capturing the satellite image is related to intensity variations [4]. Component variation in remote sensing images can happen in the absence of a light source. Intensity normalization is a preprocessing technique that modifies the histogram that balances the brightness and contrast of the image by distributing the intensity throughout the image [5].

2.3   Change Detection Algorithm

The change detection algorithm is to detect the changes in the environment by analyzing the satellite data of a particular area. There are different approaches for analyzing change detection like the Algebra-based approach, Classification-based approach, Transform-based approach, Advanced model, and GIS. One of the approaches is Algebra-based change detection where mathematical operations are applied to each image. It uses the change by threshold selection which helps in finding out the changes in the area of an image. The distribution of intensity shows in the histogram which contains a single peak that determines the threshold for change detection [6]. The objective of those algorithm is to discover the changes in a particular region at different times.

2.4   Image Segmentation

Image segmentation is a process of dividing the pixels of an image into multiple clusters/groups. Image partitions have some correlated portions like the image's color or texture in each region or cluster. An image is divided into different sets of clusters in this segmentation process. Some pixel characteristics like color or intensity are similar in each cluster are similar [7]. In the case of a grayscale image, the brightness or intensity is used for segmentation. For a clearer visualization, the image pixels need to be isolated properly in image processing, this is called image segmentation. Its application is to identify forest cover, urban areas, disaster-prone areas, and so on.

There are three different approaches

- Find the regions based on the intensity levels.
- Threshold selection is based on the distribution of pixel properties, such as if intensity value of pixel is greater than threshold then change otherwise no change.
- Finding the regions directly.

Region-based methods are based on dividing the entire image into sub-regions depending on some rules like all the pixels in one region must have the same graylevel. A common pattern of intensity values is found in one region or cluster. The cluster is referred to as the region/group, the regions according to their functional roles are the goal of the image segmentation.

2.4.1   Threshold-Based Segmentation

Threshold-based segmentation is the method of image segmentation where the pixels of an image changes depending on the threshold value, pixel values change to analyze the results. The color images are 3D arrays because red, green, and blue values are

needed to be stored. These values are integers from 0 to 255. The color image is transformed to a grayscale image and further transformed into a binary image. Segmentation combines thresholding technique with manual selection of threshold [8]. Clustering is another region-based segmentation technique that presents a clustering algorithm based on optimization [9]. The advantage of this technique results in good segmentation in less computational time. A parallel image segmentation technique is proposed in which uses threading (parts of images) and k-means clustering [10].

2.5 Clustering

Clustering is a data reduction tool that groups the data into clusters or groups that have certain similarities between data that are easier to manage than big data. Machine learning, image processing, and data mining are the area where clustering acts as an important task. The words clustering and classification are two different terms clustering is an unsupervised learning process, whereas classification is a supervised learning process. The efficiency of clustering will increase with low inter group or cluster similarity and high intra group or cluster similarity [11]. K-means is one of the clustering algorithm. The algorithm was proposed by Mac Queen in the year 1967 [12]. It was introduced to solve various clustering problems. The algorithm aimed at grouping data into k clusters based on randomly selected initial center points. The clustering is done by reducing the Euclidean distance between the data items and the center points. The input is the set of data points from d_1 to d_n and for k-means we need to find the number of clusters, where k is taken as the input which tells how many clusters we need to find out. Let the target object is d, d_i is the center point of ith cluster C_i, the function is as follows:

$$E = d(d, d_i) = \sum_{i=1}^{k} \sum_{x \in C_i} |d - d_i|^2$$

Color-based segmentation using k-means clustering is to segment colors using the $L*a*b*$ color space and k-means clustering [13]. "Luminosity layer '$L*$', chromaticity layer '$a*$' shows where color falls along the red-green axis, and chromaticity layer '$b*$' showing where the color falls along the blue-yellow axis" [14, 15]. The algorithm steps are:

Step 1: Read the image.

Step 2: Convert RGB image to L*a*b color space.

Step 3: Classification using k-means clustering.

Step 4: Display the clustered image.

The selection of threshold value to determine the change detection in term of area. In this technique there is a conversion of grayscale map to binary form and the threshold point will determine the change area or no change area. The selection of threshold is a very critical, low threshold removes the area to be segmented where as high threshold falsely segments few areas. Hence, balancing the threshold is important for better segmentation [16].

3 Accuracy Measurement

The binary maps represent the change detection results; the white pixels give the changed area whereas the dark pixels give the unchanged areas. The obtained result is accurately assessed for deciding the result of the considered problem. The main problem lies in collecting temporal remote sensing data, which causes difficulty in accuracy assessment. There are several accuracy assessment techniques like error matrix or the confusion matrix for change detection analysis. The error matrix helps in better understanding the difference between the ground truth image and the results obtained as a section of the segmentation procedure. The accuracy parameters are the Producer's accuracy and the User's accuracy [17]. A good change detection obtains, the overall accuracy value should be higher [18].

4 Discussion and Results

The input of the proposed model is taken from a part of Chandaka Wildlife Sanctuary, Odisha. Satellite images of a small part of Chandaka forest are collected from Google Earth Pro taken from historical imagery. This forest has the study of deforestation over few years till now. Analysis of the images taken from 4 years of the same region to measure the change detection. The images taken as dataset has height (rows) of 2782 pixels and width (column) of 4800 pixels. Figure 2 shows the satellite images of a part of Chandaka forest from different time period (a) year 2014 (b) year 2016 (c) year 2018 (d) year 2020.

The color images are 3D arrays because it stores red, blue, and green values (RGB) for each pixel that range from 0 to 255. For better visualization the convert the images into grayscale images. In grayscale images only one value is represented that is intensity, so the images are stored in 2D arrays. Figure 3 shows the histogram after intensity adjustment of image of the year 2021. The X axis represents the intensity values and Y axis represents the pixel count. The image is clustered in a single group because the brightness and contrast are not spread equally throughout the image. The function imadjust () will spread the pixel by maintaining the original shape for a better intensity range. The intensity of the images is examined and adjusted by taking intensity values and pixel count into the account. The images will be brighter and
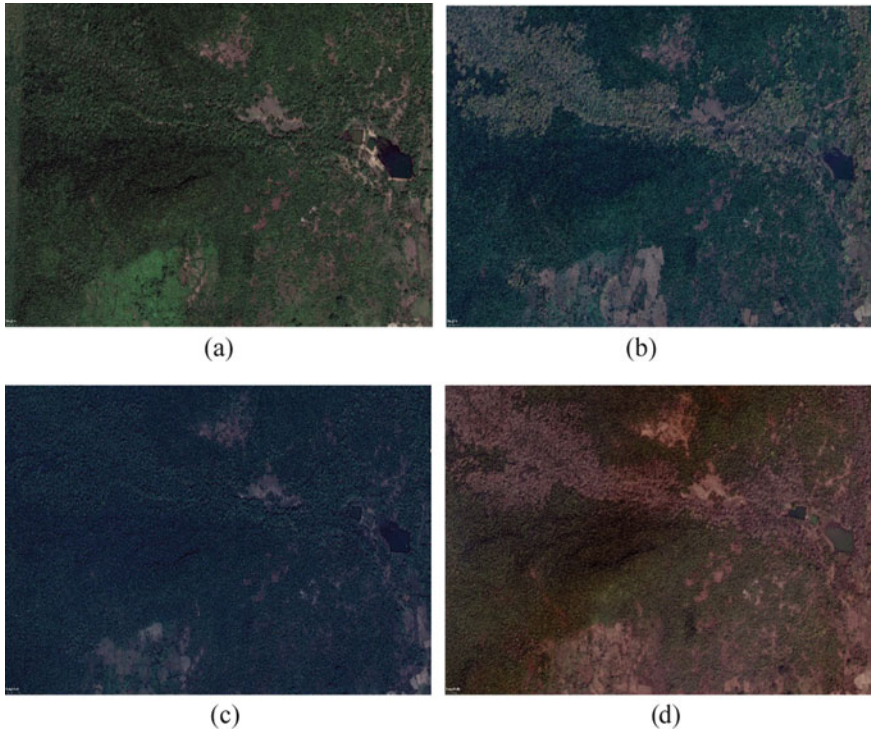
Fig. 2 Satellite images a region of Chandaka forest, **a** 2014, **b** 2016, **c** 2018, **d** 2020

intensity values will spend out through; contrast will also increase for better image processing.

Intensity adjustment is done maintaining the original shape. For a brighter change detection, intensity needs to be adjusted manually. Lower threshold isolates the deforested areas and falsely segment some areas that are not deforested and high threshold removes the segmentations by missing some areas that are deforested. Now a threshold needs to be set, here 173 threshold value is picked for better change detection.

(I)   Intensity value > Threshold, deforested

(II)   Intensity value < Threshold, not deforested

The digital image contains the discrete small picture elements called as pixels. Each pixel of an image contains a value according to the intensity called as digital number (DN). Figure 4 shows the image of year 2020 and digital number of the marked location [19]. It is dependent on intensity level which ranges from 0 to 255 and also upon the electromagnetic energy received by the sensor. The size of pixel affects the reproduction of the details in digitized image. For preservation of details
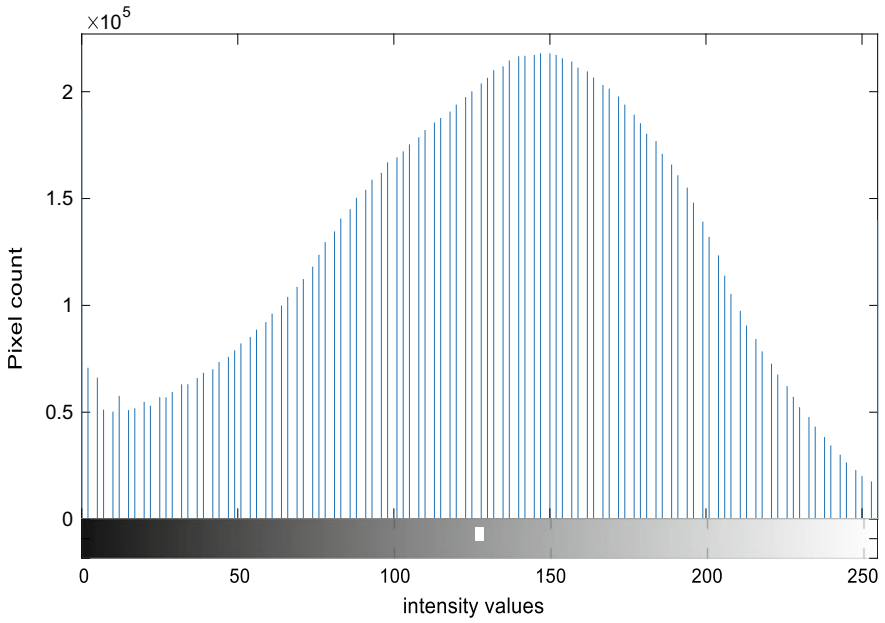
Fig. 3 Histogram after intensity adjustment of image of the year 2021
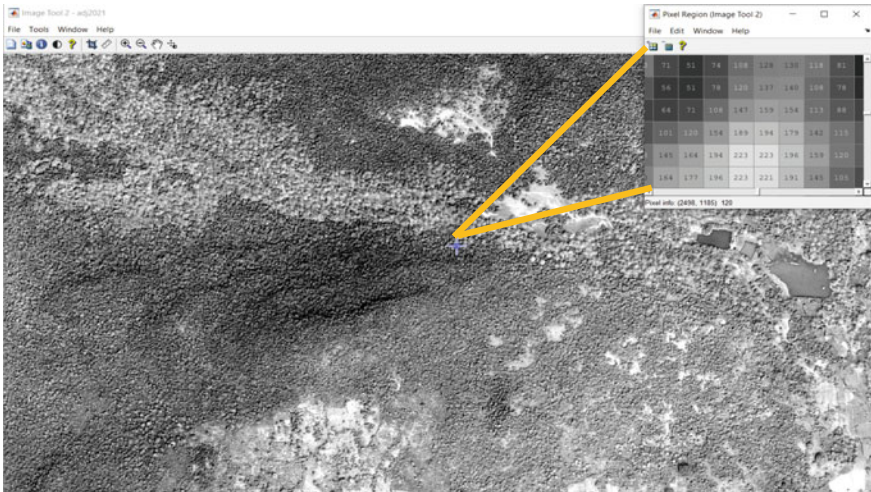
Fig. 4 Image of year 2020 and digital number

and proper digital representation of scene smaller pixel is essential. Information is lost, when there is zooming of image beyond a certain limit and there is appearance of pixels only. Using a digital image processing algorithm, the digital numbers representing their intensity level in an image may be displayed.

In the segmentation process all the images from different time needs to be processed. Same threshold also segments most of the area but need to be checked again. Few areas and scale and text also falsely segmented due to brightness of the image. In this case intensity might not be used for segmentation but color can be used for segmentation. Then segmentation can be done using the parameter color. HSV where Hue determines the color, saturation determines the shade and value determines the brightness. The color is segmented or removed by adjusting Hue parameter. Text and scale in the image are removed using value parameter, then that image is converted to binary image. Satellite image with the binary image after segmentation is shown in Fig. 5 from the year 2014 and 2020.

There is a considerable jump between 2018 and 2020, clearly visible from Fig. 6. The segmented region expanded from the year 2014 to year 2020, but in the year 2020 area increased drastically as shown in Fig. 7. Area of the deforested region is calculated in pixel and sqm^2 presented in Table 1.

Segmentation can be done by k-means using k-means clustering. The clustered images of year 2020 are shown Fig. 8 in 3 clusters. In Fig. 9, the original image and segmented image is shown for clear visualization of the segmentation.
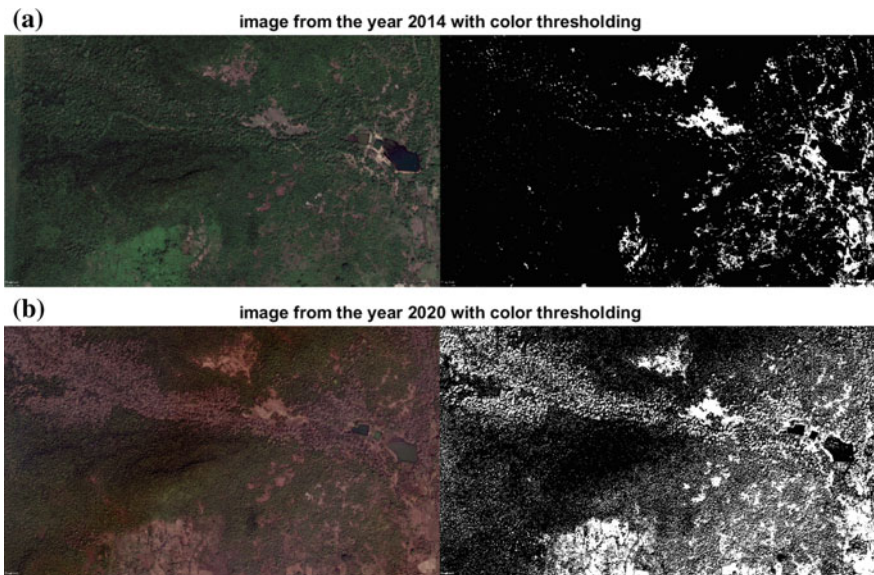
Fig. 5 Satellite image with the binary image after segmentation, **a** 2014, **b** 2020
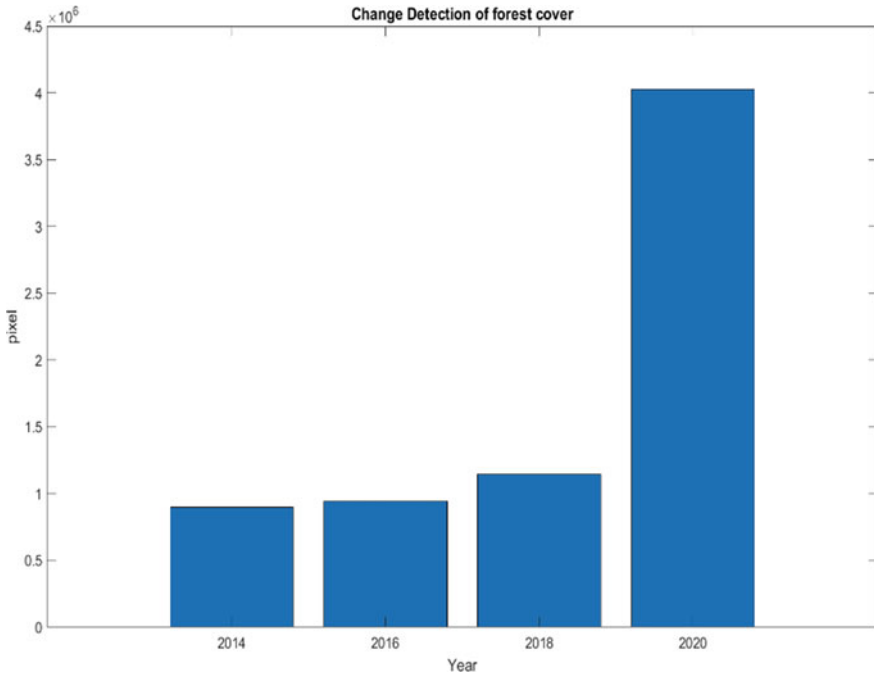
Fig. 6 Bar chart representing segmented images over different years 2014, 2016, 2018, 2020

| (a)2014 | (b)2016 | (c)2018 | (d)2020 |
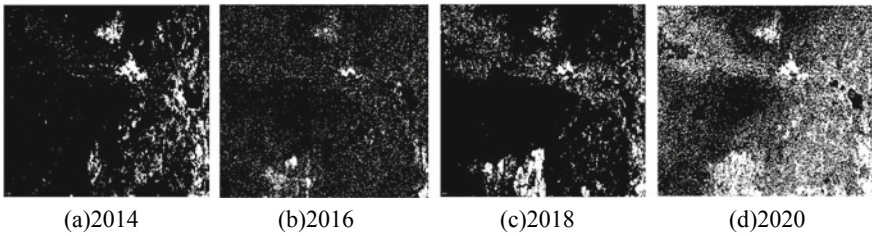
Fig. 7 Binary images showing segmented areas, **a** 2014, **b** 2016, **c** 2018 and **d** 2020

Table 1 Area of segmented regions

Year	BW	Area in pixel	Area in sq. m^2
2014	2782 × 4800	899,337	354,684
2016	2782 × 4800	944,269	372,405
2018	2782 × 4800	1,145,369	451,715
2020	2782 × 4800	4,030,245	1,589,464
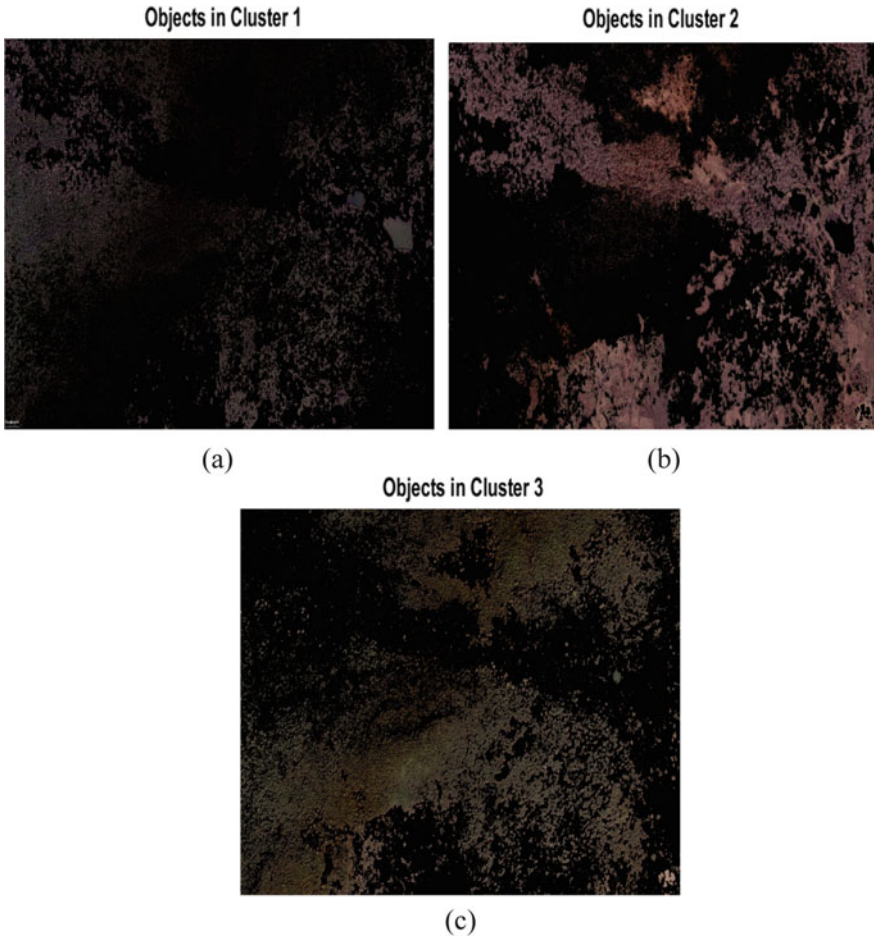
Fig. 8 Color segmentation using k-means, **a** cluster1, **b** cluster2, **c** cluster 3
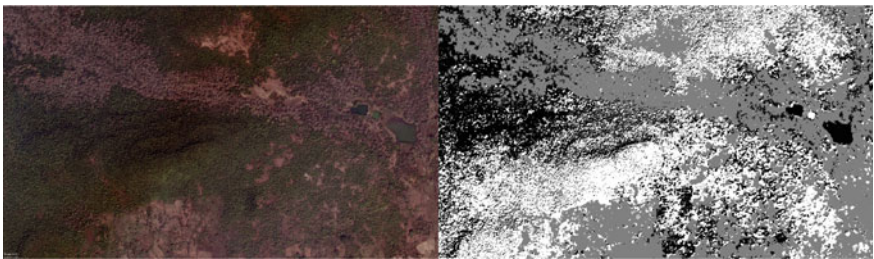
Fig. 9 Image segmentation by k-means

5   Conclusion and Future Scope

In the above study a clear result comes out as degradation of forest areas in Chandaka Wildlife Sanctuary in the year 2020 compared to 2014 due to human interaction and natural calamities. Remote sensing application is very important for satellite image processing. The challenges can be with the large image size and the noise filtration on the satellite images. In the future, deep neural network out of various applications can be implemented for change detection, to improve the change detection accuracy. A hybrid framework can also improve the accuracy of change detection. For a larger view of the area, GIS will be implemented for change detection.

References

1. Debata S, Swain KK (2020) Mammalian fauna in an urban influenced zone of Chandaka-Dampara Wildlife Sanctuary in Odisha, India. J Threatened Taxa 12(8):15767–15775
2. Rout S, Rout DK, Mohanty TL, Kar MR (2020) Forest land use map of Chandaka wildlife sanctuary, Odisha using multidate data. J Pharmacogn Phytochem 9(3):05–10
3. Anju Asokan JA (2019) Change detection techniques for remote sensing applications: a survey, Earth Sci Inform 12:143–160
4. Hao M, Shi W, Deng K (2016) Feng Q Superpixel-based active contour model for unsupervised change detection from satellite images. Int J Remote Sens 37:4276–4295
5. Wan X, Liu J, Li S, Dawson J, Yan H (2018) An illumination-invariant change detection method based on disparity saliency map for multitemporal optical remotely sensed images. IEEE Trans Geosci Remote Sens 99:1–14
6. Ferraris V, Dobigeon N, Wei Q, Chabert M (2018) Detecting changes between optical images of different spatial and spectral resolutions: a fusion-based approach. IEEE Trans Geosci Remote Sens 56:1566–1578
7. Morad AH, Al-Dabbas HM (2020) Classification of brain tumor area for MRI images. In: 1st international conference on pure science (ISCPS). J Phys: Conf Ser 1660. IOP Publishing. https://doi.org/10.1088/1742-6596/1660/1/012059
8. Chouhan S, Kaul A, Sharma U (2018) Soft computing approaches for image segmentation. Multimed Tools Appl 77:28483–28537
9. Bose S, Mukherjee A, Madhulika, Chakraborty S, Samanta S, Dey N (2013) Parallel image segmentation using multi-threading and kmeans algorithm. IEEE Int Conf Comput Intell Comput Res 1–5
10. Kapoor S, Zeya I, Singhal C, Nanda SJ (2017) A grey wolf optimizer based automatic clustering algorithm for satellite image segmentation. In: 7th International conference on advance computer communication, ICACC-2017, vol 115, pp 415–22
11. Davies DL, Bouldin DW (1979) A cluster separation measure. IEEE Trans Pattern Anal Mach Intell 1(2):224–227
12. Fouss F, Pirotte A, Renders JM, Saerens M (2007) Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. IEEE Trans Knowl Data Eng 19(3):355369
13. Chitade AZ, Katiyar SK (2010) Color-based segmentation using K-means clustering. Int J Eng Sci Technol 2(10):5319–5325
14. Juang L-H, Wu M-N (2010) MRI brain lesion image detection based on color-converted K-means clustering segmentation. Measurement 43(7):941–949
15. Sa M, Shettya A (2015) A comparative study of image change detection algorithms in MATLAB. In: International conference on water resources, coastal and ocean engineering (ICWRCOE). Aquatic Procedia 4:1366–1373

16. Paul GC, Thomas CR (2006) Characterisation of mycellal morphology using image analysis. In: Advances in biochemical engineering/biotechnology book series ABE, vol 60
17. Thakkar AK, Desai VR, Patel A, Potdar MB (2017) Post-classification corrections in improving the classification of land use/land cover of arid region using RS and GIS: the case of Arjuni watershed, Gujarat, India. Egypt J Remote Sens Sp Sci 20:79–89. https://doi.org/10.1016/j.ejrs.2016.11.006
18. Sadeghi V, Farnood Ahmadi F, Ebadi H (2018) A new fuzzy measurement approach for automatic change detection using remotely sensed images. Meas J Int Meas Confed 127:1–14
19. Introduction to Remote Sensing, NCERT https://ncert.nic.in/textbook/pdf/kegy307.pdf

Mental Stress Detection Using GSR Sensor Data with Filtering Methods

Ramesh K. Sahoo, Alok Ranjan Prusty, Ashima Rout, Binayak Das, and Padmini Sethi

Abstract Study of the stress level in the human body is vital now a days. It is very important to assess the mental state of the human being with significant physiological changes. Proper and on time diagnose of the stress and anxiety may make one's lifestyle happier, healthier, and more productive. Persons, when stay and work far from their places; undergone many types of life changes and become the victim of stress, trauma, and anxiety. Hormonal changes in the human body due to stress can be reflected in terms of physiological and psychological changes. It becomes more significant to address such situations at remote places by analysing physiological data and send the same data through heterogeneous wireless communication for further analysis. In this paper, it has been identified three different activities with varied positions and sending of galvanic sensing response sensed data to the intended sink node through the heterogeneous wireless communication medium. Galvanic sensing response sensed data are different in respect to the contact surface area with the body, body position, environment, and activities. Proper investigation of sensed data can give real time solution.

R. K. Sahoo (✉) · A. Rout
IGIT Sarang, Sarang, Odisha, India
e-mail: ramesh0986@gmail.com

A. R. Prusty
DGT, NSTI (W), RDSDE, Kolkata, West Bengal, India

B. Das
TCS Bhubaneswar, Bhubaneswar, Odisha, India

P. Sethi
RDW University, Bhubaneswar, Odisha, India

1   Introduction

Stress is one of our body's response. Stress is a sensation of inexplicable anxiety, anguish, and fear from which people are not able to free themselves. Many kinds of situations or life events can create stress. Stress of human being is the collective reaction to a stimulus in terms of physiological, cognitive, and behavioural reactions [1, 2]. When human body meet stress, internal systems of our body produces stress hormones lead to psychological sweating. Psychological sweating due to emotive stimuli is more prominent in the body parts like palm, face, and sole [3]. Human body is distributed with three kinds of sweat glands with distinct function. Eccrine gland is one of them present on palm. This glands on palm are not susceptible to heat rather to mental stress and emotional anxiety [4]. Different psychological and physiological activities that create serious problem and affecting both mental and physical health of the people [5]. This requires proper analysis and observation of the stress level. This becomes more essential for the human beings present at the remote locations.

Stress turns to be mental or physical sickness like extensive anxiety depression and tiredness. Physiological variations have a keen relationship related with mental sickness. This may be related with various circumstances and age groups, which may affect the performance, behaviour, thinking ability, responsibility, etc. [6]. Sensing these variations, it can be possible to shape a system which at times identify stress in terms of bodily or mental responses of distinguishing patterns through numerous sensor signals. In these types of variations, stress makes the glands more active. This is because mind and physical body are interrelated [7, 8]. Physical sickness may turn to be to mental anxiety that may have negative impact on human health conditions. Physiological data can be well suited to identify whether a person is under stress or not. It is possible to forecast enhancingor reducing intensities of stress level once a person is involved in a some kind of activity. In a experimental sense, the electrical resistance of the skin may be used to find and observe the level of stress [9]. When eccrine gland becomes heavy with fluids, the skin with higher conductance value bring the fluid to the outer surface of the body part in terms of sweat. The conductance can be constantly recorded by putting the two electrodes on the finger. Galvanic Skin Response (GSR) [10] is a vital physiological and psychological measure, which can be used for monitoring the mental and physical stress of a body. Skin conductance level in GSR keeps track of the changes in sweat gland activity. It is a sign of physiological or psychological stimulation and electrodermal response.

In this paper, it has been shaped up a prototype using GSR [10] sensors connected to wireless media to monitor the degree of mental and physical stress in human body. The communication media are made heterogeneous [11] with the use of Wi-Fi and GSM shields for communicating between the nodes. The sensed analogue data converted into text data are sent via wireless setting towards the destination node in the network. These text data are converted into graphs through graphical interface and used for analysis at the destination stations. The sensingdata may contain noise

due to external interference which will be highly required to eliminate. This can be done by filtering mechanism.

Related work has been discussed in Sect. 2. The methodology presented in Sect. 3, followed by result and analysis, conclusion are shown in Sects. 4 and 5, respectively.

2   Related Work

GSR sensor technology is catching popularity and interest amongst the development and research community, especially in development of wearable that deals with human activity and emotion recognition. Various kinds of devices are available to measure the GSR at the various parts and places of the human body [12]. Electrodermal response of bio-medical system can translate electrical characteristics of the skin based on physiological and psychological activities of a living human being through skin conductance response to measure human emotions [13, 14]. Stress can be induced based on the different activities performed by the body and GSR value will vary due to this change of the human body activities and movements [15]. Finding appropriate ways to sense GSR bio signals is a substantial contribution for the science, engineering, and research to keep on working in new and advanced approaches to expand quality of human life [12].

Activities performed at different levels in daily lifestyle, can also be boosted with the support of sports and other physical doings [16]. Spontaneous detection and classification of physiological activity of human being can be very useful to notify and alert the user about his routing work performed and encourage him for more active lifestyle [17]. A personalization procedure and an effective decision tree classifier have been applied on wireless motion detection bands and PDA is discussed by the author in [18]. This method is outputting an individual's stress pattern, relating to the behaviour of the human body under the varieties of conditions with varied degrees of stress. GSR response is examined to define the stress level of the body. The information gathered from GSR sensor or the speech of a person helps in drawing conclusion in deciding the accurate classification techniques, studied in [19]. The stress measurement can be performed using fuzzy logic with two physiological sensors GSR and Heart Rate (HR) sensor signals are very useful in recognizing the stress pattern in human beings [20]. The monitoring of the stress through the different wireless communication media like; Wi-Fi, Bluetooth, ZigBee, GSM, etc. of human body in remote places is highly essential now-a-days. Remotely monitoring of dual parameters such as the heart rate and patient body temperature in wireless sensor network [21] and mobile-based augmented reality are described in [22]. In [23], the authors have described the analysis of mental stress using Wi-Fi communication mode in wireless environment. The data have been captured in remote places and need to send to the destination station, analysed and again to send the result back to source point.

All the above studies have used single communication media. Hence, the system becomes more prone to the failure on communication device, malfunctioning results

sensed data become unreachable in the network system. So, it would be better to
have the provision of multiple communication media for the network system imple-
mented at remote places. Therefore, it has been proposed the heterogeneous wireless
communication media using two communication modules, i.e. Wi-Fi and GSM shield
through a node in the wireless sensor network. İn this kind of environments, more
noise may be occured. To deal with noise, various filtering methods may be used.

3   Methodology

In this section different components that are used for this experiment using
heterogeneous wireless communication media has been presented.

3.1   Data Collection in Wireless Medium

Arduino Uno, Wi-Fi Shield, GSM shield, e-Health Sensor board, and GSR Sensor
are the different components considered for collection and communication of data
in wireless network environment.

The GSR sensing method is one of the most popularly used skin conductance
response method obtained using skin conductivity that can be recorded at fingers
of the palm particularly middle and index. It can also be used at different parts of
the different fingers of the palm. The outcome of the measurement is better when
used at middle parts of middle and index fingers of the palm. The GSR detects and
computes the psychological galvanic skin conductance of the human being. It can also
be considered for research in physiological or emotional arousal. GSR Sensor has
been associated with e-Health board (Shield V2.0) equiped with various sensors [24]
as observed in Fig. 1, enables the Arduino Uno [25] to make different kinds of medical
and biometric applications for a living human body. Various health sensors can be
employed for observing and examining various physiological activities performed
by living human body through e-Health board and Arduino Uno microcontroller that
is a microcontroller-based mother board which can be connected to an electronic

Fig. 1 GSR sensor with
e-health sensor board [24]
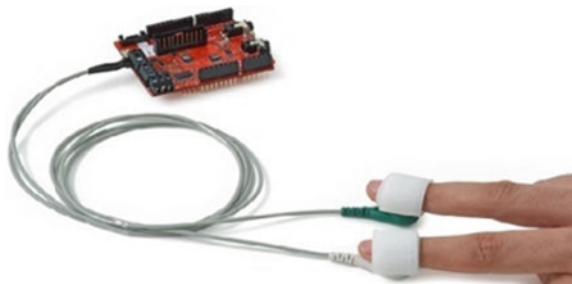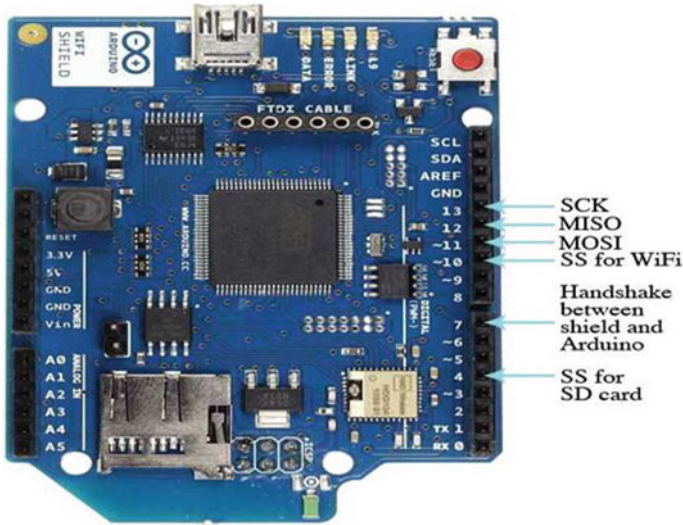
Fig. 2 Wi-Fi shield for arduino [25]

device such as laptop, desktop with a USB cable or Wi-Fi. It is powered by AC-to-DC adapter or a suitable battery to start the Arduino Uno board. Arduino Uno mother board has UART TTL (5 V) serial pins, which is on digital 0 and 1. The value 0 is for RX and 1 for TX. The Arduino software enables the sensing data to be forwarded or received from the Arduino Uno microcontroller over serial monitor. It is programmable using Arduino sketch to perform sensing task using various sensor and the result will be displayed on the serial monitor of the system. Sensing data or information obtained using various sensor can be transmitted wirelessly through Wi-Fi and/or GSM shields for any real time application.

The connection between Arduino Uno mother board and wireless network has been established through Arduino Wi-Fi Shield as shown in Fig. 2 by using the 802.11b and 802.11g. An Atmega 32UC3 is used to provide the network accomplishment using UDP and TCP protocol. The Arduino Wi-Fi Shield provided with a slot for micro-SD card, to store sensing data for serving over the wireless network and it can be well-placed with Arduino Uno and Arduino Mega mother board. Arduino Uno mother board can be connected with Internet through GPRS wireless network using Arduino GSM Shield as observed in Fig. 3 and also can use voice calls or SMS. Radio modem M10 by Quectel has been used by GSM Shield to link with the board through AT commands. Pin 2 and 3 will be connected with TX pin and RX pin, respectively, for software serial communication with the M10.

Modem Reset (pin 7)

GSM TX (pin 3)
GSM RX (pin 2)
Not connected
(pins 0 & 1)
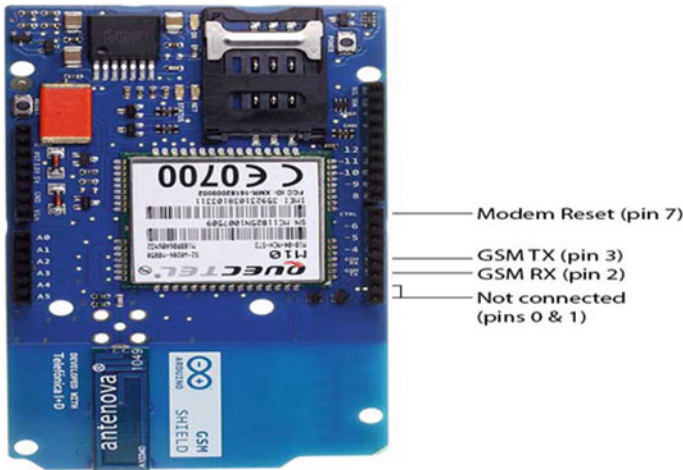
Fig. 3 GSM shield for arduino [25]

3.2 Experimental Setup

The hardware components such as Arduino, e-Health board, Wi-Fi Shield, GSM Shield, and GSR Sensor are used for the experimental setup. The experiment runs through the software components like; Matlab and Arduino IDE. The e-Health board is linked with Arduino and performed its operation based on embedded Arduino software in Arduino Uno board. It is communicated to an electronic device such as: laptop, desktop or server through wireless mode using Wi-Fi shield and GSM shield for data communication.

The connection between the Arduino Wi-Fi Shield and Arduino Uno board provides wireless connectivity for the data acquisition node. The received data at the source node has been linked with SD card available in Arduino Wi-Fi shield for data storage. Further the Arduino GSM shield is connected with the data acquisition node, by placing it over Arduino Wi-fi Shield. The node can be shown as per Fig. 4. Metallic electrodes will be attached at the middle position of the index and middle finger using velcro. It continuously receives physiological sensor data from the sensor node which is attached with it. Two kinds of communication media such as, Wi-Fi and GSM are used to communicate between source and destination nodes in the wireless sensor network. There may be possibility of intermediate nodes between source and destination nodes. When Wi-fi network connectivity is not available then Arduino GSM shield is used to transfer received physiological sensor data to the sink node in wireless environment.

The sensed data can be continuously added in the SD card in text format and further the collected data are to be sent to the sink node through the wireless network via either Wi-Fi shield or GSM shield. Further analysis of data received at the destination node will be done by the specialist doctor or analyzer in the remote locations. The

experimental setup of e-Health Sensor board based on Arduino Uno with Wi-Fi and
GSM shield is presented in Fig. 4 and collection and observing nodes are shown in
Fig. 5.

In the data acquisition node, sensed data has been collected by GSR sensor, and
stored in csv format in text file in the SD Card of Wi-Fi shield. This saved data in
the csv formatted text file can be transferred to any sink node in the wireless sensor
network, where the node can be a laptop, desktop, smart phone, any other remote
device etc., through the Wi-Fi shield of the node. If Wi-Fi connectivity is not available
at that place it sends the data using GSM shield of the node. The data stored in text
file can be converted into graphs through the Matlab or KST of Arduino. Matlab is
used to plot the graph in this paper.

Data has been collected from various person for the various kinds of activities
in different emotional behaviour such as exercise, tension, and normal in dissimilar
positions such as: standing, sitting, and lying for a time duration of ten minutes with
115,200 baud rate.
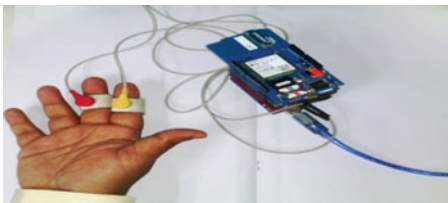
(a) Data acquisition Node   (b) Observing Node

Fig. 5 Data acquisition and observing node setup

4 Experimental Results

The results for different physical positions have been obtained and described in this part of the paper for various physical positions such as lying, sitting, and standing in different moods such as normal, tension, and exercise. The data received is further analysed.

From Fig. 6, it is difficult to analyse the data using raw data. It may include noise with raw data. So, it is required to eliminate the raw data using different filters. Here, it is considered low pass filter and sliding average window filter for better analysis. Between two methods it has been observed that low pass filter provides better performance as compared to sliding average window. Low pass filters is useful for outputting a smoother form of a signal, eliminating the small variations. The detail has been displayed in Fig. 7. Figure 8, shows different level of data for different moods at laying position of a body during 10 min. In Fig. 9, it has been observed that the values are in similar during constant strain of a body at sitting position. The body generates GSR values without fluctuation at standing position with constant tension
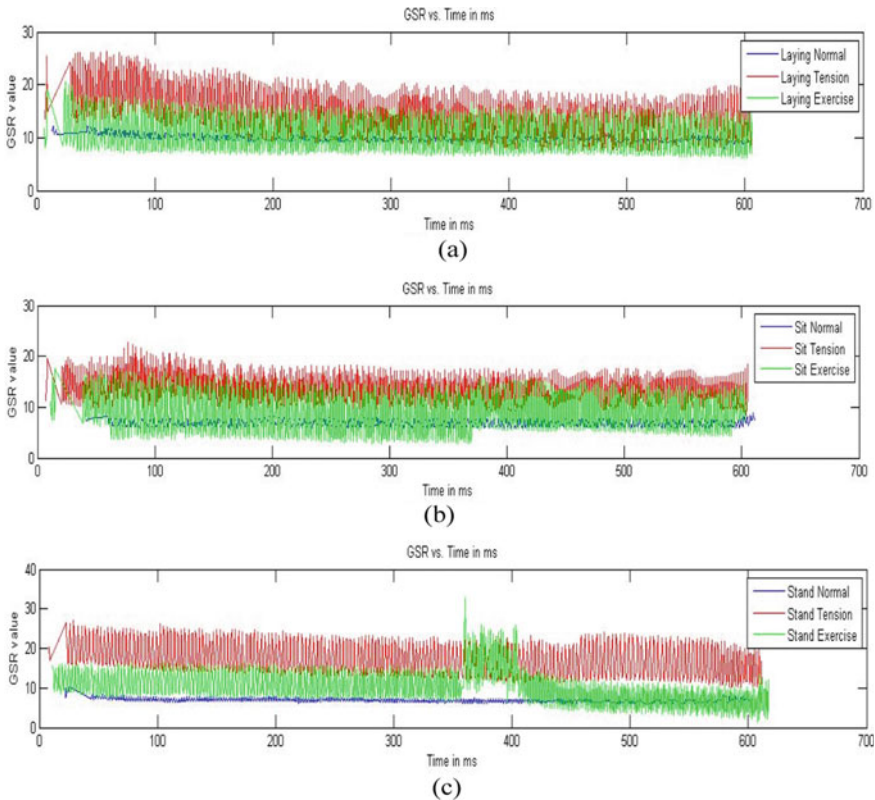
Fig. 6 GSR value versus time in ms for standing position in normal, tension, exercise moods [26]
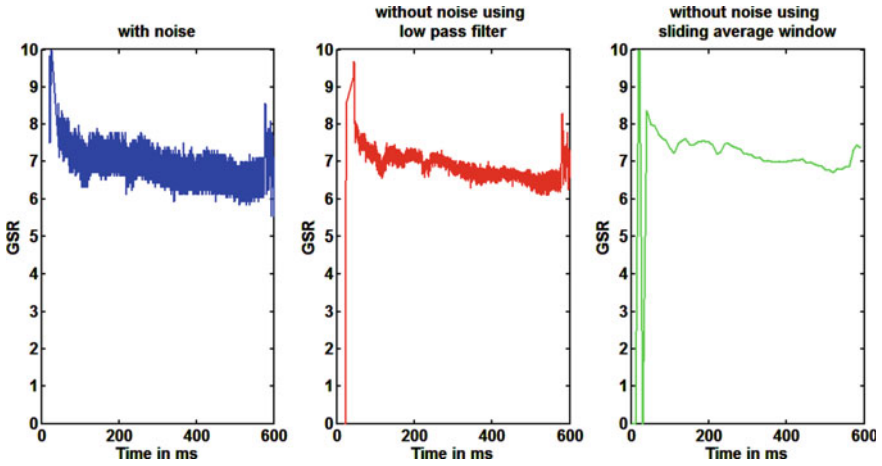
Fig. 7 Results with and without filtering the data
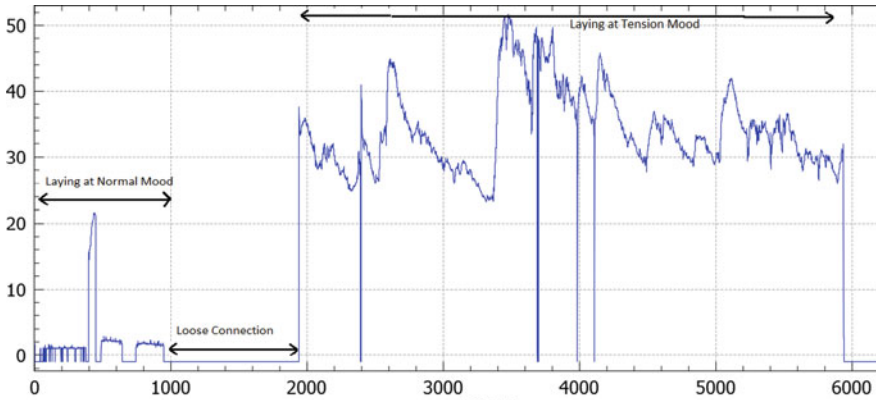
Fig. 8 Various form of data at laying position of a body
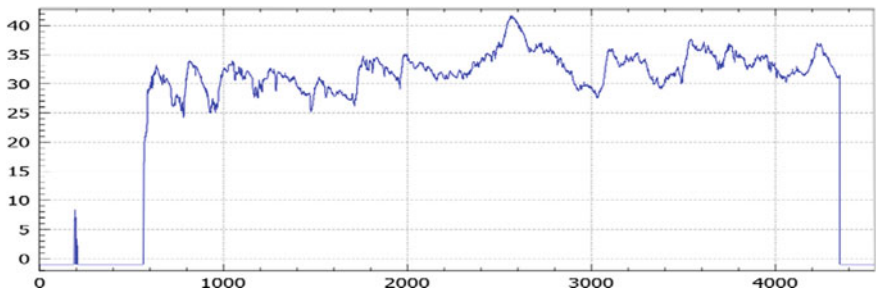
Fig. 9 Data at sitting position of a body with tension mood from starting to end period of time
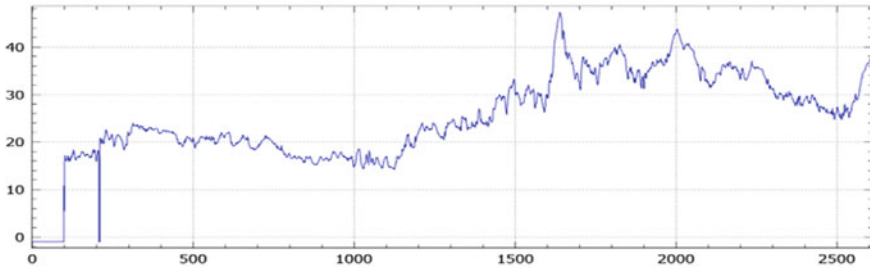
Fig. 10 Data at standing position of a body with tension mood from starting to end period of time

of a body, which is shown in Fig. 10. Particularly the value is negative if the GSR sensor points are not connected with skin.

All the sensing data (in text) format will be transmitted to the end node via heterogeneous wireless media linked with e-health board and connected with Arduino Uno mother board. The received information at the destination point can be analysed by expert Doctor(s)/analyser(s). The analysed results can be provided to the source remote location.

5   Conclusion

The objective of this study is to build a prototype to analyse the stress level of human body in remote locations using heterogeneous wireless communication media. The heterogeneity is brought into picture with the use of Wi-Fi plus GSM shield along with e-health sensor platform and connected with Arduino Uno board. The physiological data are collected using GSR sensor for distinguished bodily positions such as laying, sitting, and standing with changed mental conditions like, normal, tension, and during the exercise. The above observations presented in this paper may be helpful to study the state of the mind in terms of different emotions and also the stress level of the human body in real time. This system can be helpful for the treatment and counselling of those people who are under any kind of mental stress and they stay away from the city location, where doctors or health care facilities are not available.

References

1. https://www.mentalhealth.org.uk/a-to-z/s/stress. Retrieved 21.10.2021
2. Goumopoulos C, Mentib E (2019) Stress detection in seniors using biosensors and psychometric tests in proceed. In: International conference on pervasive computing advances and applications (Percaa 2019), Proceedia-2019
3. Harker M (2013) Psychological sweating: a systematic review focused on aetiology and cutaneous response. Skin Pharmacol Physiol 26:92–100. https://doi.org/10.1159/000346930
4. Asahina M, Poudel A, Hirano S (2015) Sweating on the palm and sole: physiological and clinical relevance. Clin Auton Res 25:153–159. https://doi.org/10.1007/S10286-015-0282-1
5. Altini M, Julienpenders, Vullers R, Amft O (2013) Combining wearable accelerometer and physiological data for activity and energy expenditure estimation. In: Proceedings of the 4th Conference on Wireless Health, Usa
6. Glanz K, Schwartz M (2008) Stress, coping, and health behavior health behavior and health education: theory, research, and practice, pp 211–236
7. Vijaya PA, Shivakumar G (2013) Galvanic skin response: a physiological sensor system for affective computing. Int J Mach Learn Comput 3(1) (2013)
8. Juha P, Miikka E, Panu K, Janim,Johannes P, Ilkka K (2006) Activity classification using realistic data from wearable sensors. Ieee Trans Inf Technol Biomed 10(1):119–128
9. Navea RF, Buenvenida PJ, Cruz CD (2019) Stress detection using galvanic skin response: an android application. J Phys: Conf Ser 1372:012001. https://doi.org/10.1088/1742-6596/1372/1/012001. Iop Publishing
10. Westerink JHDM, Van Den Broek EL, Schut MH, Van Herk J, Tuinenbreijer K (2008) Computing emotion awareness through galvanic skin response and facial electromyography. Probing Exp Philips Res 8:149–162
11. Wang C, Li S, Li B, Chen Q (2020) An asynchronous data transmission method in heterogeneous wireless networks. Complexity 2020, Article Id 8828794, 7 pages. https://doi.org/10.1155/2020/8828794
12. Sanchez-Comas A et al (2021) Correlation analysis of different measurement places of galvanic skin response in test groups facing pleasant and unpleasant stimuli. Sensors 21(12):4210. https://doi.org/10.3390/S21124210
13. Sahoo R, Sethi S (2014) Functional analysis of mental stress based on physiological data of GSR sensor. Aisc Series of Springercsi-2014, Hyderabad, India
14. Bakker J, Pechenizkiy M, Sidorova N (2011) What's your current stress level? Detection of stress patterns from GSR sensor data. In: 11th international conference on data mining workshops (Icdmw). Ieee Computer Society Washington, DC, USA, pp 573–580
15. Fenz WD, Epstein S (1967) Gradients of a physiological arousal of experienced and novice parachutists as a function of an approaching jump. Psychomat Med 29:33–51
16. Miikka E, Juha P, Panu K, Jani, Ilkka K (2008) Detection of daily activities and sports with wearable sensors in controlled and uncontrolled conditions. Ieee Trans Inf Technol Biomed 12(1):20–26
17. Juha P, Miikka E, Mark G (2010) Automatic feature selection and classification of physical and mental load using data from wearable sensors. In: Proceedings of the 10th Ieee International Itab Conference Corfu, Greece, 2–5 Nov 2010
18. Juha P, Luc C, Miikka E (2010) Personalization algorithm for real-time activity recognition using pda, wireless motion bands, and binary decision tree. Ieee Trans Inf Technol Biomed 14(5):1211–1215
19. Kurniawan H, Maslov AV, Pechenizkiy M (2013) Stress detection from speech and galvanic skin response signals. In: Ieee 26th international symposium on computer-based medical systems (Cbms), pp 209–214
20. De Santos Sierra A, Avila CS, Bailador Del Pozo G, Guerra Casanova J (2011) Stress detection by means of stress physiological template. Nature and biologically inspired computing (Nabic), pp 131–136

21. Prusty AR, Sethi S, Nayak AK (2016) A hybrid multi-hop mobility assisted heterogeneous energy efficient cluster routing protocol for wireless ad hoc sensor networks, pp 265–280
22. Cornelio F, González J et al (2014) Smart multi-level tool for remote patient monitoring based on a wireless sensor network and mobile augmented reality. Sensors, pp 17212–17234
23. Sahoo R, Sethi S (2015) Remotely functional-analysis of mental stress based on gsr sensor physiological data in wireless environment. Aisc Series of Springer India-2015, Kalyani, India
24. http://www.cooking-hacks.com/documentation/tutorials/ehealth-biometric-sensor-platform-arduino-raspberry-pi-medical
25. http://arduino.cc/en/main/
26. Sethi S, Sahoo RK (2020) Design of wsn in real time application of health monitoring system. Virtual and mobile healthcare: breakthroughs in research and practice. Igi Global, pp 643–658

Enhancing the Automated Diagnosis System of Soft Tissue Tumors with Machine Learning Techniques

P. Kowshika, S. Mousika, P. Divya, K. Lalitha, A. Jeevanantham, and H. Muthukrishnan

Abstract In our body structure, Soft Tissue Tumor (STT) is the form of tumor that might be found in tissues coupled with body cells. These tumor cells have light frequency and maintains great diversity within the body and when observed in Magnetic Resonance Imaging (MRI), appeared as heterogeneous because of this nature. When diagnosis takes place, the confusion arises with other diseases such as lymphadenopathy, struma nodosa, fibroadenoma mammae, which leads to some harmful effects when patients undergoing medical treatment. To classify these tumors, many researchers have come up with different machine learning prototypes to address the misdiagnosis problem. Evaluation of many such tumors mostly not considered about the data size and the heterogeneity. Hence, an approach is proposed based on machine learning which combines the data preprocessing for transformation of features, resampling techniques for eliminating the bias, instability deviation and classifier test performance based on decision tree (DT) in addition to support vector machine (SVM) techniques. The test results produced a better improvement when compared to the previous works like SVM, KNN and improves the accuracy rate by 90.4% in J48.

Keywords Tumor types · Soft tissue tumors · Preprocessing techniques · Support vector machine · Decision tree

P. Kowshika · S. Mousika · P. Divya · K. Lalitha (✉) · A. Jeevanantham · H. Muthukrishnan
Department of IT, Kongu Engineering College, Perundurai, India
e-mail: klalitha.it@kongu.edu

P. Kowshika
e-mail: kowshikap.18it@kongu.edu

S. Mousika
e-mail: mousikas.18it@kongu.edu

P. Divya
e-mail: divyap.18it@kongu.edu

1   Introduction

The expression "delicate tissues" alludes to tissues that help, associate, or encompass different designs and organs in the body like fat, muscles, veins, profound cutaneous tissues and nerves and also tissues encompassing the body joining's. According to the type proposes, they are touchy tissues which are influenced by a few diseases, include growths that can grow anyplace in the human body. The harmful sorts of these cancers, otherwise called Soft Tissue Sarcomas (STS), are gathered on the grounds that they share numerous minuscule components, show similar side effects, and are comparatively treated [1].

However, compelling determination of Soft Tissues Tumors (STT) is as yet a major test attributable to the trouble in identifying these malignancies. A few procedures have in this way been created to reinforce the identification of such diseases, including Magnetic Resonance Imaging (MRI) examination. X-ray is presently viewed as the standard indicative instrument for the recognition and arrangement of STT with all around described organic properties, for example, cell beginnings and growth examples used to recognize cancers [2].

X-ray can be utilized to break down textural attributes or other less described cancer qualities (normal MRI signal force, state of growth limits) for a very long time: simplicity of calculation textural attributes, (2) wide range connection of textural qualities to growth pathology and (3) heartiness to changes in MRI procurement boundaries, for example, changes in the cancer picture and defilement of the MRI picture because of field heterogeneity. Thus there is an expanding utilization of machine learning calculations to break down MRI pictures all adequately and consequently identify malignant growths. It has turned into a fundamental device for present day medication today and has been fortified by prescient programmed learning calculations that work on the indicative exhibition of existing master frameworks. Among these numerous applications, we have fostered an AI-based strategy for the auto identification and determination of cancers like STT [3].

2   Related Works

The size and the cancers can shift incredibly starting with one patient then onto the next. Moreover, the unpredictable, non-uniform growth, which represent a critical test, particularly when conventional analysis techniques rely upon clear cancer limits. Furthermore, complex MRI information for cancers from clinical investigations or manufactured data sets are hard for doctors to decipher. The MRI gadgets and conventions utilized for imaging estimating can change starting with one sweep then onto the next, forcing power inclination and different minor departure from each unique picture cut in the dataset [4].

The STT (delicate tissue growth) dataset involves attributes, including names, with mathematical and five absolute credits as summed up. The dataset contains invalid

credits that require uncommon preprocessing before order with ML calculations. The STT dataset comprises of information from the investigation of patients, among which patients were chosen. Using this dataset, the recognizable proof of the tissue growth can be effectively distinguished [5].

Rustam et al. [6], has proposed the series that contains distributions on hypothesis, applications, and plan strategies for Intelligent Systems and Intelligent Computing. For all intents and purposes all disciplines like designing, inherent sciences, PC and data science, ICT, financial aspects, business, online business, climate, medical service and life science are used to categorize the patient records.

Stephane et al., has proposed to represent another information displaying approach for five normal arrangement calculations to streamline. A bank tended to, information on its customers, items and social-monetary properties including the impacts of the monetary emergency. A unique arrangement of the 150 components has been investigated and the 21 provisions are held for the proposed approach including mark. This paper presents another displaying approach that pre-processed independently each kind of provisions and standardize them to advance forecast execution. The outcomes acquired for five most realized AI strategies are: Logistic regression (LR), Naïve Bayes (NB), decision trees (DT), support vector machines (SVM) and artificial neural network (ANN) which further developed exhibitions for this load of calculations as far as exactness and f-measure [7].

Zahras et al. [8] has proposed a significant part in assisting clinical staffs with tackling those clinical issues. Delicate tissue growths are cancers in the musculoskeletal framework that include delicate. Zhang et al. [9], has proposed the Tumor orders which are turned into a fundamental piece of present day oncology and, for pathologists, they give rules which work with indicative and prognostic reproducibility. Apparently the four most huge applied advances are: (i) the conventional acknowledgment with morphological harmless sores; (ii) the overall acknowledgment that most sarcomas can be definitively sub classified and that representing close to 5% of grown-up delicate tissue sarcomas; and (iii) the expanding increase that not exclusively known from which cell type(s) most delicate tissue cancers begin.

Komura et al. [10], has proposed the various level association of chromatin is known to connect with assorted cell capacities; nonetheless, the exact components and the 3-D design still need not set in stone. With ongoing advances in high-throughput cutting edge sequencing (NGS) procedures, genome-wide profiling of chromatin structures is made believable. Paper additionally addresses the natural importance of chromatin association and how the mix with different procedures uncovers the hidden components. This paper highlights the fundamental upgrades of as of now accessible techniques to propel comprehension of chromatin pecking order. The survey unites the examinations of both higher and essential request chromatin designs, and fills in as a guide when picking fitting trial and computational techniques for evaluating chromatin hierarchy.

3 Support Vector Machine (SVM)

In AI, SVM are directed for the related calculation of learning calculations that examine the information for arrangement and setback the examination. SVMs are the most powerful forecast method, being founded on measurable learning structures or VC hypothesis. Non-probabilistic two fold straight classifier SVM maps preparing guides to focuses in space in order to boost the width of the hole between the two classifications.

4 Proposed Methodology

In software engineering, a pre-processor (or pre-compiler) is a program that measures its feedback information to deliver yield that is utilized as contribution to another program. Information preprocessing can allude to control or dropping of information before it is utilized to guarantee or upgrade execution, and is a significant stage in the information mining measure [11]. The expression "trash in, trash out" is especially relevant to information mining and AI projects. The strategies used for the information gathering are frequently approximately can controlled, coming about in an range esteems, unthinkable information mixes and missing qualities. Information planning and sifting steps can take impressive measure of preparing time. Instances of information preprocessing incorporate cleaning, occasion choice, standardization, one hot encoding, change, highlight extraction and determination and so forth.

The result of information preprocessing is the last preparing set. Information preprocessing may influence the manner by which results of the last information handling can be deciphered. This viewpoint ought to be painstakingly viewed as when translation of the outcomes is a central issue, such in the multivariate handling of compound information (chemometrics). STT are harmful cancers that create inside tissues like muscles, fat, stringy tissues, nerves and veins. More importantly, the conflicting MRI pictures make it hard for doctors to decide a successful treatment. Furthermore, STT can undoubtedly be mistaken for different infections, for example lymphadenopathy, and strumanodosa. This symptomatic disappointment fundamentally affects the patient treatment measure. As indicated by the hypothesis referenced by the Karanian et al. [12], there are four classifications of connections-based growth development harmless injuries, cancers with neighborhood potential. At the point when a sub-atomic inconsistency of a substance had been distinguished, the meaning and that element, which is both histological and sub-atomic is acquired. Based on the current test, in this way, how to successfully utilize the attributes of these oddities for better designated treatment for soft tissue tumors.

The J48 calculation is a managed ML technique utilized for information grouping. The main role of utilizing the SVM calculation is to precisely order the imperceptible information by limiting the misclassification utilizing a choice capacity. It is finished

via preparing the SVM calculation on the preparation information and afterward utilizing the prepared to foresee the yield class new information.

Algorithm 1 J48 Algorithm

1.	**Start**
2.	Create a Root Node M
3.	IF(category(T==C)
4.	{
5.	Leaf_node=M;
6.	Mark M as C_class;
7.	Return M;
	}
8.	For i= 1 to M
	{
	Calculate infor_gain and entrophy (AI);
	}
9.	T_A = test attribute;
	M.T_A = highest infor_gain attribute;
	If (M.T_A == continuous)
	{ threshold to be calculated;}
10.	For(each T splitting)
	If(T==0)
	{child node of M=leaf node;}
	Else
	{child node of M=dtree_T;}
	Calculate classification error rate(M);
	Return M;
11.	**End**

A SVM-based methodology was used to characterize STT (SOFT TISSUE TUMOR) and also non-STT growths. SVM calculation utilizes a few boundaries, for example, the kernel.J48 calculation gives preferred exactness over the SVM and in blunder rate is low. The J48 calculation is utilized to order various applications and perform precise after effects of the characterization. J48 calculation is one of the most mind-blowing AI calculations to analyze the information completely and persistently as shown in algorithm1. At the point when it is utilized for example, it consumes more memory space and exhausts the exhibition and exactness in arranging clinical information. Our proposed strategy is to quantify the further developed presentation and produce higher pace of precision.

A SVM-based methodology was applied to characterize STT (SOFT TISSUE TUMOR) and non-STT growths. For doing SVM calculation utilizes a few boundaries, for example, the kernel.J48 calculation gives preferred exactness over the SVM and in blunder rate is low. The J48 calculation is utilized to order various applications and perform precise after effects of the characterization. J48 calculation is one of the most mind-blowing AI calculations to analyze the information completely and persistently. At the point when it is utilized for example, it consumes more memory space and exhausts the exhibition and exactness in arranging clinical information. Our proposed strategy is to quantify the further developed presentation and produce higher pace of precision.

5 Data Preprocessing

The preprocessing step changes various sorts of elements into a particular configuration diminishing the preparing time and further developing the arrangement execution while staying entirely stable against factors scales by managing each trait independently. To plainly show these various advances, let us consider an illustration of occasions which contains some normal provisions accessible in significant client datasets.

6 Gray Image

After the preprocessing step, the information picture is changed over into the dim scale picture to get the most elevated level of distinguished ability of the tissue cancer. Gray Level Co-event Matrices (GLCM) is one of the earliest texture analysis strategy utilized for content extracted from the image. The gray image was investigated properly with the context of image retrieval with pixels. The overall flow of the work is shown in Fig. 1.

7 Results Drawn

7.1 Experimental Setup

In the past segment, we examined the presentation of the model by looking at SVM and the J48-DT algorithms. In this part, the outcomes and the calculations are compared to bring the performance of the past work. In existing works, the calculations make a worldwide conversation of the contribution of existing work contrasted with the best in class programmed similar review of performance consequences of
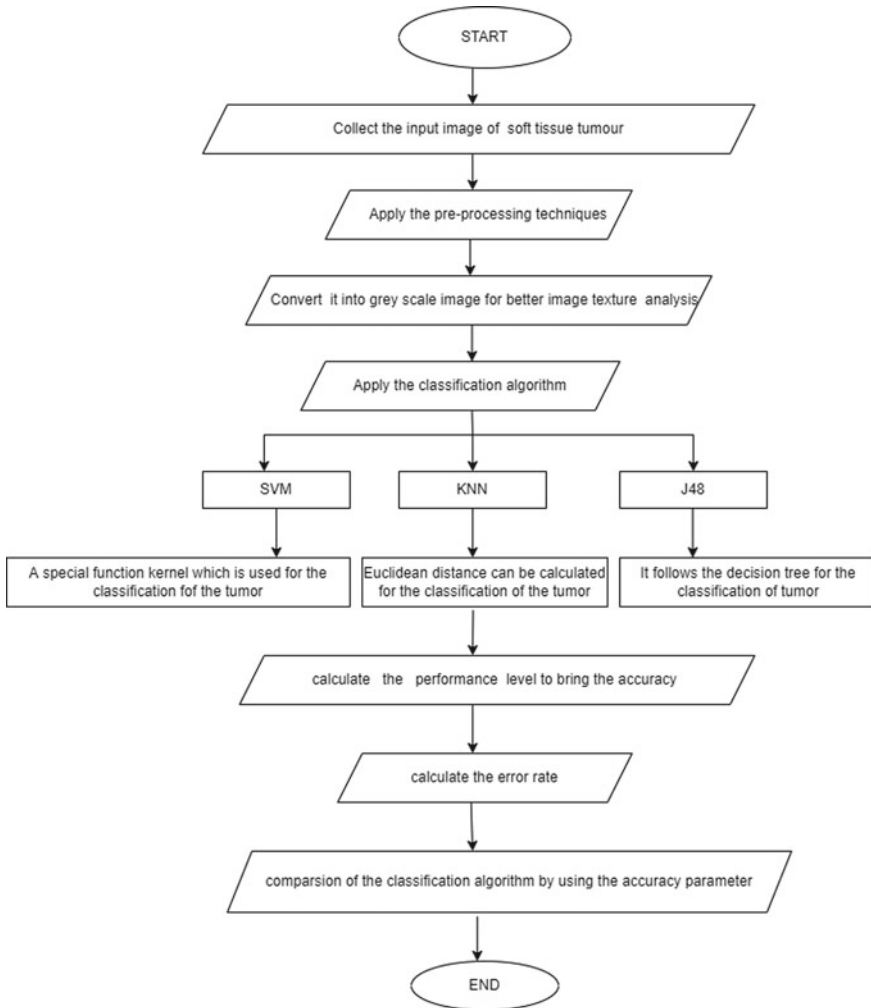
Fig. 1 The overall flow of the process

two models considered here with more established ML calculations, gives less preci-sion. It could be evidenced that SVM and DT extraordinarily surpass the algorithm and better performance of different models for the three measures (accuracy, AUC and f1-measure).
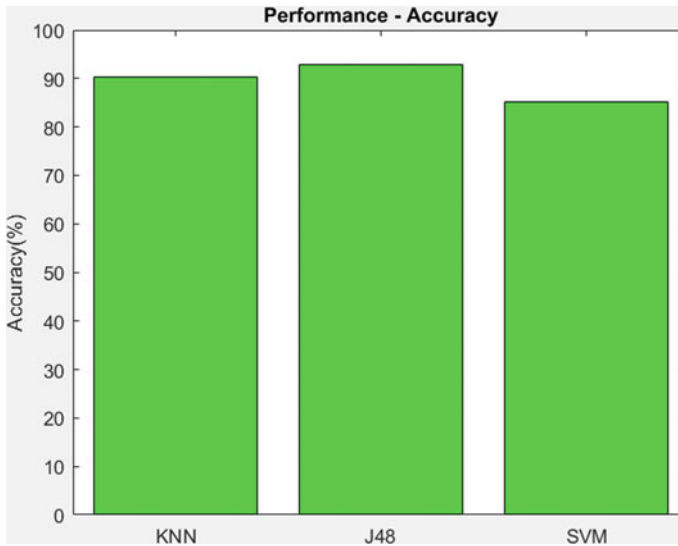
Fig. 2 Performance accuracy

7.2 Performance Accuracy

After gray image texture analysis, feature transformation is applied to convert the categorical data in terms of numerical values. Order precision is just the pace of right arrangements, either for a free test set or utilizing some variety of the cross-approval. A specific exactness might appear to be acceptable from the beginning, another cancer classifier model that consistently predicts harmless would accomplish precisely the same precision as such, this model is no more excellent than one that has zero percent capacity to portion the tissue growth. The accuracy rate for SVM is 85%, KNN is 90% and J48 is 92%. As shown in Fig. 2.

7.3 Error Rate

In advanced transmission of tumors, the quantity of the touch blunders is the quantity of got pieces of an information stream over a correspondence channel that have been modified because of clamor, obstruction, twisting or spot synchronization mistakes. The piece blunder rate (BER) is the quantity of touch mistakes was calculated on unit time. The error rate achieved for SVM is 14%, KNN is 9% and J48 is 7% is clearly given in Fig. 3.
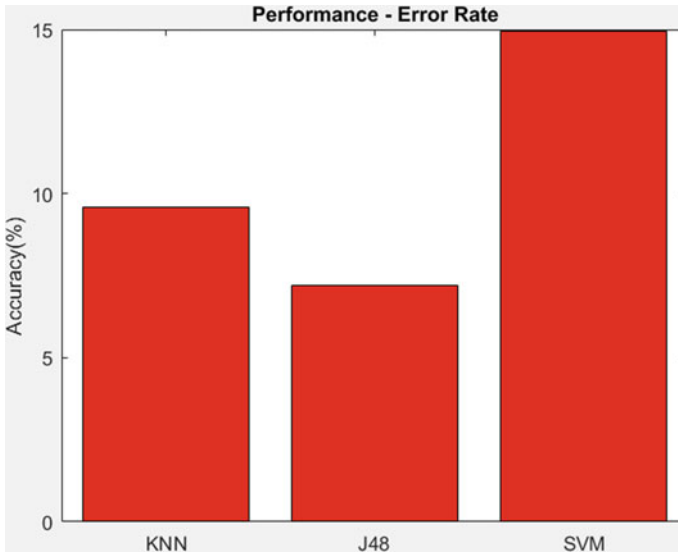
Fig. 3 Error rate

8 Conclusion

High accuracy estimations advanced by ML calculations can help applications in many clinical field. These devices have made it conceivable to work on the presentation of PC helped indicative frameworks essentially as of late, and their reconciliation keeps on giving a test to current well-being foundations. In the wake of incorporating another information preprocessing strategy, two best classifiers considered, specifically SVM and J48 algorithms. This examination showed that regardless of whether the DT calculation is somewhat more productive than the SVM calculation, the J48 model is substantially more delicate to the quantity of factors than the SVM model. The execution of a PC supported symptomatic framework dependent on our model could likewise end up being more proficient and viable than the analysis completed altogether by a visual appraisal done by a radiologist and best in class models frequently investigated from test pictures. In future, error rate will be zero on reinforcing and ceaselessly working on our model by joining powerless calculations to adjust it to the programmed analysis of different sorts of infections like glaucoma.

References

1. Dadar M, Pascoal T, Manitsirikul S, Misquitta K, Fonov V, Tartaglia M, Breitner J, Rosa-Neto P, Carmichael O, Decarli C, Collins D (2017) Validation of a regression technique for segmentation of white matter hyperintensities in Alzheimer's disease. IEEE Trans Med Imaging 36:1758–1768
2. Martinez-Murcia F, Ortiz A, Gorriz J, Ramirez J, Castillo-Barnes D (2020) Studying the manifold structure of Alzheimer's disease: a deep learning approach using convolutional autoencoders. IEEE J Biomed Health Inform 24:17–26
3. Kamalam GK, Lalitha K, Priyadharshini E (2021) Forecasting the spread of COVID-19 using supervised machine learning models. AIP Conf Proc 2387:140017
4. Sruthi P, Butchi Raju K (2021) Covid-19 forecasting using supervised machine learning techniques—survey. E3S Web Conf 309:01218
5. Nie L, Zhang L, Meng L, Song X, Chang X, Li X (2017) Modeling disease progression via multisource multitask learners: a case study with Alzheimer's disease. IEEE Trans Neural Netw Learn Syst 28:1508–1519
6. Zhou T, Liu M, Thung K, Shen D (2019) Latent representation learning for Alzheimer's disease diagnosis with incomplete multi-modality neuroimaging and genetic data. IEEE Trans Med Imaging 38:2411–2422
7. Abirami T, Lalitha K, Jayadharshini P, Madhuvanthi T (2021) Future: HCI in public conveyances. AIP Conf Proc 2387:140042
8. Lalitha K, Poongodi C, Anitha S, Vijay Anand D (2021) An energy-efficient routing with particle swarm optimization and aggregate data for IOT-enabled software-defined networks. Lecture Notes in Networks and Systems. 185 LNNS, pp 105–114
9. Lalitha K, Thangarajan R, Poongodi C, Anand D.V (2018) Sink originated unique algorithm for clustering and routing to forward aggregated data in wireless sensor networks. In: IEEE international conference on proceedings, I2C2SW 2018, pp 127–130, 8997141
10. Kumar S, Fred A, Varghese P (2018) Suspicious lesion segmentation on brain, mammograms and breast MR images using new optimized spatial feature based super-pixel fuzzy C-means clustering. J Digit Imaging 32:322–335
11. Razek A (2020) Editorial for "preoperative MRI-based radiomic machine-learning nomogram may accurately distinguish between benign and malignant soft tissue lesions: a two-center study. J Magn Reson Imaging 52:883–884
12. Zhong W, Zhang F, Huang C, Lin Y, Huang J (2021) Classification of clear cell renal cell carcinoma based on tumor suppressor genomic profiling. J Cancer 12:2359–2370

A Moth Flame Optimization for Failure Aware Controller Placement in Software-Defined Networks

Sagarika Mohanty [ORCID]**, Bibhudatta Sahoo, Srinivas Sethi, Hemant Kumar Apat, and Khushboo Kanodia**

Abstract Software-defined networking (SDN) is an upcoming network model that emphasizes the separation of the control plane and the data plane, resulting in more flexibility, programmability, and network management. The problem of placement of controllers deals with the count of controllers that are needed and their location in the network to maintain the network structure. In a large network where multiple controllers are needed, the quantity and position of the controllers have a major impact on SDN's performance and reliability. The network performance degrades significantly in case of controller failure. To avoid this situation, planning is required. A moth flame optimization algorithm is proposed for controller placement with planning, CPWP_MFO and compared with a genetic algorithm CPWP_GA. The aim is to place *m* controllers in the network to reduce the worst-case and average propagation latency considering controller failure and capacity constraint. The simulation results show that CPWP_MFO yields better results than CPWP_GA.

Keywords Software-defined network (SDN) · Controller placement with planning (CPWP) · Controller failure · Moth flame optimization (MFO) · Propagation latency

S. Mohanty (✉) · B. Sahoo · H. K. Apat
National Institute of Technology, Rourkela, Rourkela, India
e-mail: sagarikam_23@yahoo.com

B. Sahoo
e-mail: bdsahu@nitrkl.ac.in

H. K. Apat
e-mail: hemantkumar_apat@nitrkl.ac.in

S. Sethi
Indira Gandhi Institute of Technology, Dhenkanal, India
e-mail: srinivas_sethi@igitsarang.ac.in

K. Kanodia
Nokia Networks, Bangalore, India

1 Introduction

Software-defined networking (SDN) is an upcoming network paradigm that separates the control and data planes to allow for more flexible network management and innovation. In SDN, the forwarding devices form the infrastructure layer, the control functionality presents in the controllers form the control layer and business applications constitute the application or management layer [1]. Placing a single controller in a network is advantageous and gives a global view of the system. But this raises issues such as scalability and single controller failure. Thus, multiple controllers are required. Finding an ideal controller number and their placement is known as the controller placement problem (CPP) [2].

Placement problem is a NP-hard problem since placing m-controllers among n possible locations $\binom{N}{m}$ where $m < n$ [3]. The performance will be affected by randomly placing controllers in the network. In a network like SDN, failure can occur due to various reasons such as controller failure, link failure, or both. In case of controller failure, the switches are unplugged from their primary or initial controller and need administrator's intervention to be allocated to a different active controller with sufficient controller capacity. The outcome will be sudden increase in latency and consumption time. This requires proper planning. For every switch if there is the availability of a primary and a backup controller then in case of failure the switch will be automatically reallocated to its backup controller without administrator's interference. This paper follows this approach of Killi and Rao [4] where the problem is described as a mixed-integer linear program with worst-case latency as a performance measure. We have extended their work of controller placement with planning (CPWP) in this study and considered average and worst-case latency as our main objectives. Due to the large size and the complexity of the network, a heuristic algorithm is proposed considering single controller failure at a time. The main contributions are summarized below:

– A moth flame optimization algorithm is proposed for controller placement with planning (CPWP_MFO) considering single controller failure.
– CPWP_MFO is compared with the genetic algorithm (CPWP_GA).
– The objective is to place m controllers in the ideal location to reduce the worst-case and average propagation delay. This includes the delay from a switch to its primary controller and from the primary controller to its closest controller on occurrence of controller failure considering controller's capacity.
– CPWP is compared with capacitated controller placement (CCP).
– Three network datasets from topology zoo are considered for evaluation.

The proposed CPWP_MFO avoids the local optima problem unlike CPWP_GA and has high exploration and exploitation which outperform other algorithms such as CPWP_GA as shown from the results.

The remainder of the article is organized as follows. The next section entails an outline of the relevant work. Section 3 contains the problem formulation. Section 4

describes the methodology and description of CPWP_MFO and CPWP_GA.break Section 5 contains the analysis and results. Finally, the article's summary and possible future studies are described.

2 Related Work

Various research work considered failure while considering placement of controllers. In [4], authors illustrated a MILP approach for resilient placement of controllers that plans ahead for controller failure to escape from disconnection, repeated interference of network operators and rapid increase in latency. The aim of the paper is to reduce the worst-case delay on occurrence of controller failure. It evaluates its performance on various topologies and found that the suggested approach performs better. Hock et al. [5] proposed a framework for the resilient placement of controllers. They suggested that more than 20% of nodes present in the network topology need to be controller for uninterrupted connection in node or link failure environment. The authors of [6] proposed a clique-based method for resilient placement of controller. Its suggested approach minimizes the cost and provides high performance. The authors also claim that failure of the controller would affect the entire system. So, planning is required beforehand to handle the failure cases. In [7], authors described two methods for reliable placement of controller in case of node and link failure. The first one is to connect the switch with two controllers, one as the primary and second as a backup controller. The second method is that a switch is connected to each controller through two disjoint paths. The authors of [8] suggested a distributed network for reliable placement of the controller. The system is divided into various subnets, and each subnet load is calculated based on the total number of nodes, its degree and packet loss rate. In [9], authors described heuristic methods for reliable controller placement considering the total average latency as a metric. They illustrated varna-based optimization and compared it with jaya, teacher learning-based, and particle swarm optimization. In [10], the authors examined propagation delay, hop count, and link utilization while allocating switches to controllers. They generalized a flow setup time model and discussed control latency in transmission of packets. In [11], the authors illustrated the problem of failures in SDN and addressed a primary backup controller mapping solution to reduce the controllers count and remapping to minimize the cost function. They develop heuristic algorithms for large networks. [12] Petale et al. illustrated a technique for controller placement in which the network will regain its normal state in reduced time during failure. The goal is to reduce the worst-case delay while placement with maximum utilization of resources for multiple link failure. [13] Aravind et al. presented simulated annealing algorithm to minimize the worst-case delay from the switch to its backup controller.

While developing solution for failure aware placement of controllers most of the literature work did not consider the load of the switches and the controller capacity. Many literature considered this as ILP problem. Very few research works are available where metaheuristic approaches are adopted for failure aware CPP.

3   Problem Formulation

The network graph is symbolized as $G = (V, L)$. $V = Z \cup X$ are the nodes of the network. $Z = \{z_1, z_2, \ldots, z_n\}$ be the set of switches, $X = \{x_1, x_2, \ldots, x_m\}$ be the set of controllers and $L = \{l_1, l_2, \ldots, l_e\}$ are the set of links.

Let $P = 2^{|V|} - 1$ describes all possible placement of controllers. Let $d(z_i, x_j)$ be the minimum distance or shortest communication path from switch z_i to controller x_j. The load of switch z_i is ρ_i and μ_j be the capacity of controller x_j. A binary variable B_{ijk} is fixed as 1 when switch z_i is assigned to primary controller x_j and backup controller x_k; otherwise, it is set as 0. Propagation latency is divided into two types: the average switch to controller delay is the average packet communication delay between switch and controller $\lambda_{sc-avgL}(P)$ and the worst-case switch to controller delay is the maximum transmission delay between switch and controller $\lambda_{sc-worstL}(P)$. These are calculated as,

$$\lambda_{sc-avgL}(P) = \frac{1}{|V|} \sum_{i \in Z} \sum_{j.k \in X} \{d(z_i, x_j) + d(x_j, x_k)\} B_{ijk} \tag{1}$$

$$\lambda_{sc-worstL}(P) = \max_{i \in Z} \min_{j,k \in P} \{d(z_i, x_j) + d(x_j, x_k)\} B_{ijk} \tag{2}$$

The aim is to reduce the average and worst-case delay which includes the delay from a switch to its primary controller and then to its backup controller in case of failure subject to some constraints.

$$\text{minimize} \quad \lambda_{sc-avgL}(P) \tag{3}$$

$$\text{minimize} \quad \lambda_{sc-worstL}(P) \tag{4}$$

Subject to,

$$\sum_{x_j \in P} \mathcal{A}_j = m \tag{5}$$

$$\sum_{x_j, x_k \in P} B_{ijk} = 1 \quad \forall z_i \in Z \tag{6}$$

$$\sum_{z \in Z} \rho_i B_{ijk} \leq \mu_j A_j \quad \forall x \in P \tag{7}$$

$$A_j, B_{ijk} \in \{0, 1\}, \quad \forall z_i \in Z, \forall x_j, x_k \in P \tag{8}$$

$\mathcal{A}_j$ is set to 1 when a controller is placed at x_j. In the network, there are a total of m controllers (5). B_{ijk} is set as 1 when switch z_i is allocated to primary controller

x_j and backup controller x_k (6). The sum of the switches load should not exceed the controller's capacity as captured by (7). The binary variables are described in (8).

In CPWP as in Eqs. (1) and (2), there is the availability of a primary and backup controller for every switch. But in capacitated controller placement (CCP), detached switches of the inactive controller are allocated to the next active controller with enough controller capacity, thereby increasing latency.

4 Algorithm Description

4.1 Moth Flame Optimization for CPWP (CPWP_MFO)

CPWP_MFO is based on moth navigation [14]. Moths are the candidate solutions, and its variables are moth positions and flames are the best position of moths. A logarithmic spiral is selected as the prominent update characteristics of moths. A spiral function is used to update the moth's position in relation to the flame.

$$mothPos_p = S(mothPos_p, flameNo_q) \tag{9}$$

S, the logarithmic spiral function is denoted as,

$$S(mothPos_p, flameNo_q) = DT_p e^{lra} \cos(2\pi r) + flameNo_q \tag{10}$$

The distance from the pth moth for the qth flame is represented by DT_p, whereas l is a constant that represents the logarithmic spiral shape and a random variable ra is between $[-1, 1]$.

DT is calculated as,

$$DT_p = |flameNo_q - mothPos_p| \tag{11}$$

It can sometimes reduce the exploitation capacity when updating the position of moths in n locations to find the optimal option. For this goal, a mechanism is proposed.

$$flameNo = round\left(flmax - iter \times \frac{flmax - 1}{miter}\right) \tag{12}$$

where $iter$ stands for the current iteration and $flmax$ is the total number of flames. $miter$ denotes the total iterations.

The shortest path matrix $srtMat$, population size $popSize$, and maximum iterations $miter$ are the input parameters to the CPWP_MFO. $srtMat$ is computed from the topology's latitude and longitude information. Individual moths represent one kind of placement. Each $mothPos$ is specified as a m-dimensional vector, with each

---

Algorithm 1 CPWP_MFO

Input: $srtMat$, $popSize$, m, $miter$
Output: $\lambda_{sc-avgl}$, σ_{bpos}
1: $srtMat \leftarrow$ shortest path matrix
2: $popSize \leftarrow$ total number of moth positions
3: $m \leftarrow$ problem dimension
4: $miter \leftarrow$ total iterations
5: **for** $iter \leftarrow 1$ to $popSize$ **do**
6:     $mothPos_i \leftarrow$ generate random moth position
7: **end for**
8: **for** $iter = 1$ to $miter$ **do**
9:     $mothScore \leftarrow$ Fitness $(srtMat, mothPos)$
10:     **for** $iter \leftarrow 1$ to $popSize$ **do**
11:         $flameNo_q \leftarrow$ update flame
    positions using Eq. (13)
12:     **end for**
13:     **if** $iter == 1$ **then**
14:         $bFlame \leftarrow$ sort$(mothPos)$
15:         $bScore \leftarrow$ fitness $(mothScore)$
16:     else
17:         $bFlame \leftarrow$ sort$(mothPos, bFlame)$
18:         $bScore \leftarrow$ sort$(mothScore, bScore)$
19:     **end if**
20:     **for** $i = 1$ to $popSize$ **do**
21:         **for** $j = 1$ to k **do**
22:             update ra
23:             update DT as per Eq. (13)
    according to the corresponding moth
24:             update $mothPos(p, q)$ as per
    Eqs. (12) and (14)
25:         **end for**
26:     **end for**
27: **end for**
28: $\sigma_{bpos} \leftarrow bFlame$
29: $\lambda_{sc-avgl} \leftarrow bScore$
30: **return** $\lambda_{sc-avgl}$, σ_{bpos}

---

dimension representing one of the controllers' location, given the number of controllers as m. Switches are allocated to each controller as per $srtMat$. Fitness for each moth is calculated using Eqs. (1) and (2). The program will run for the maximum number of times in order to obtain the optimal position σ_{bpos} and the lowest latency $\lambda_{sc-avgl}$ and $\lambda_{sc-worstl}$ considering one failure.

4.2 Genetic Algorithm for CPWP (CPWP_GA)

The input parameters to the $CPWP\_GA$ are same as CPWP_MFO. The population are randomly generated, and each chromosome's fitness function is calculated using

Eqs. (1) and (2). Here elitism method is used for selection. A single point crossover is performed on the selected two members. Then mutation is done with 0.01 probability. The program will run for the maximum number of times in order to obtain the optimal position σ_{bpos} and the lowest latency $\lambda_{sc-avgl}$ and $\lambda_{sc-worstl}$ considering one failure.

5 Result and Analysis

For implementation, three networks of different sizes are considered from the topology zoo [15]. The various networks are TataNID (143 nodes and 194 links), IRIS (51 nodes and 104 links), and AT & T (25 nodes and 56 links). MATLAB R2014a is used to find the $srtMat$ from the information available in GraphML of the network using Haversine formula. This will give the shortest distance between two nodes. Rest of the programs is written in Python. Simulation parameters are fixed for implementation after running the programs a couple of times. The controller's capacity is fixed as 7.8×10^6 packets/second. 100 kilo requests per second are set as the demand of a switch. The population size is different for the three networks. $maxitr$ is set as 100. The aim is to reduce the worst-case and average propagation delay which includes the delay from switch to its primary controller and from the primary controller to its backup controller considering controller failure.

5.1 Performance of CPWP_MFO and CPWP_GA Against Objective Function

The performance of CPWP_GA and CPWP_MFO is compared for different m values on three networks, AT & T, IRIS, and TataNID as Figs. 1 and 2. The average and worst-case propagation delay is computed as per Eqs. (1) and (2). Generally, switch to controller latency decreases along with increasing controller number. CPWP follows the same procedure and minimizing latency by allocating switches from their failed primary controller to its backup controller. From these figures, it can be easily observed that the changes in average and worst-case latency are almost linear after a certain controller number. We have considered the controller range from 2 to 9. After analyzing the results, it is observed that the performance of CPWP_MFO is better and provides good results as compared to CPWP_GA.

5.2 Comparison

Controller placement with planning for failure (CPWP) plans ahead for controller failure. Due to certain reason, when failure occurs, the disconnected switches of the
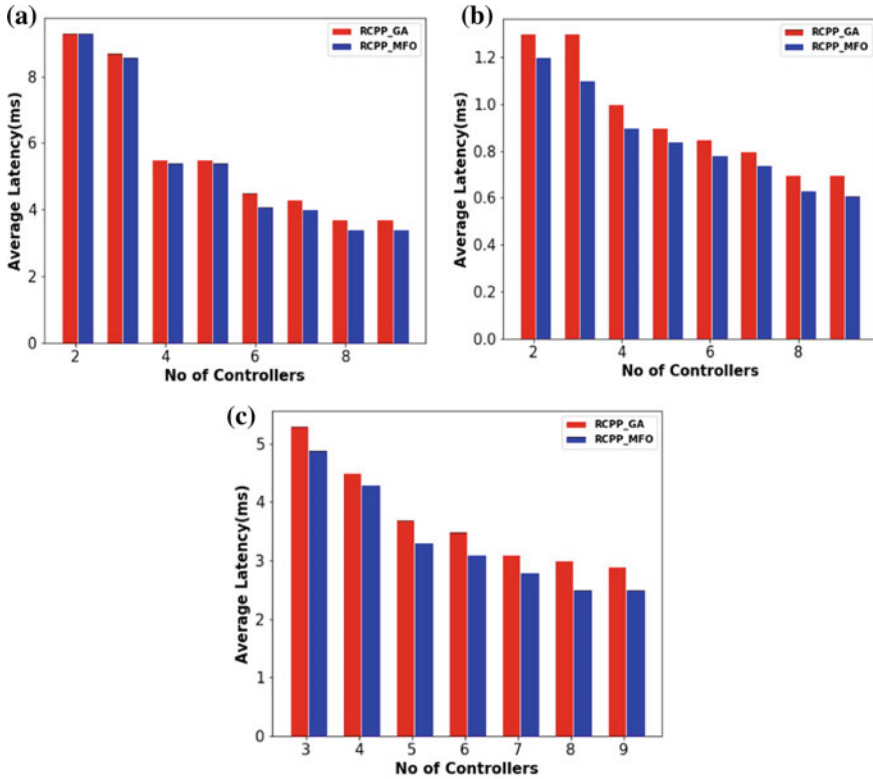
Fig. 1 Average latency of CPWP_GA and CPWP_MFO in **a** AT & T, **b** IRIS and **c** TATANID
networks

failed controller are assigned automatically to its backup controller. We have com-
pared CPWP with capacitated controller placement (CCP) in which the disconnected
switches of a failed controller are allocated to another nearest active controller by
the network operator. This work is similar to the paper of Killi and Rao [4]. They
have considered worst-case latency as their objective but we have considered both
worst-case and average case latency as shown in Fig. 3. This latency increases drasti-
cally in CCP with failure as planning has not been done beforehand. On the contrary,
as planning for failure is done in CPWP the increase in latency is not very signifi-
cant. However, for failure-free case in CCP the latency is lower compared to CPWP.
CPWP_GA and CPWP_MFO provide good results when compared to CPWP for
both average and worst-case latency. In AT & T network for 4 controllers, average
latency is around 6 ms in CPWP, whereas it is around 5.6 ms in CPWP_MFO.
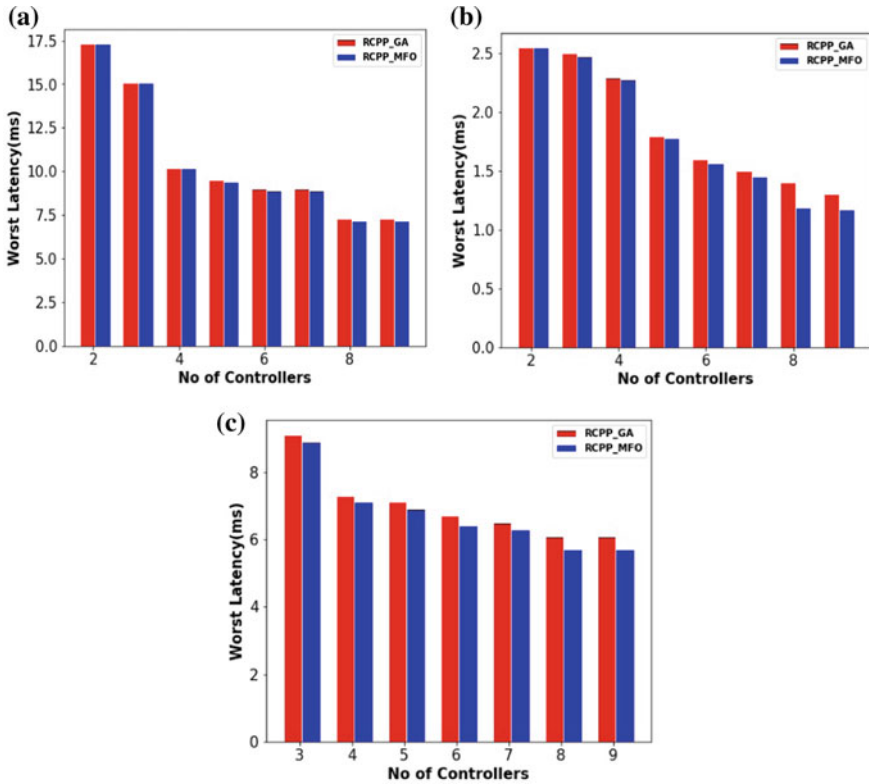
Fig. 2 Worst-case latency of CPWP_GA and CPWP_MFO in **a** AT & T, **b** IRIS, and **c** TATANID networks
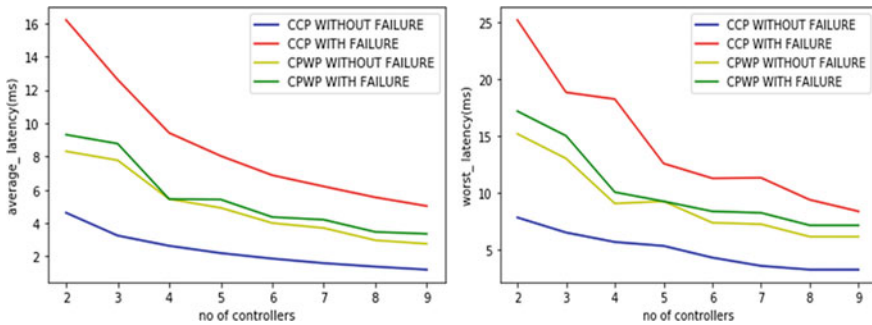
Fig. 3 Comparison of average and worst-case latency of CCP and CPWP with and without failure in AT & T network

6 Conclusions

In this study, the problem of controller placement in SDN is addressed that mitigate sudden increase in latency on occurrence of controller failure. The aim is to reduce the average and worst-case propagation delay which includes the delay from switch to its primary controller and from the primary controller (failed one) to its closest controller considering the load and controller capacity. For this, the metaheuristic solution CPWP_MFO is proposed and compared with CPWP_GA. Then evaluated on three networks of topology zoo. From the results, it can be shown that the CPWP_MFO yield better result than CPWP_GA for both worst-case and average case latency. In comparison, CPWP is compared with CCP without and with failure for both average and worst-case latency in AT & T network. Due to proper planning in CPWP, there is no drastic increase in latency on occurrence of controller failure. We intend to investigate controller failure in the future research at more than one level considering load balancing of controllers.

References

1. Das T, Sridharan V, Gurusamy M (2020) A survey on controller placement in SDN. IEEE Commun Surv Tutor 22(1):472–503, First quarter. https://doi.org/10.1109/COMST.2019.2935453
2. Kumari A, Sairam AS (2019) A survey of controller placement problem in software defined networks. arXiv preprint arXiv:1905.04649
3. Heller B, Sherwood R, McKeown N (2012) The controller placement problem. ACM SIGCOMM Comput Commun Rev 42(4):473–478
4. Killi BPR, Rao SV (2016) Controller placement with planning for failures in software defined networks. In: 2016 IEEE international conference on advanced networks and telecommunications systems (ANTS), pp 1–6. https://doi.org/10.1109/ANTS.2016.7947795
5. Hock D, Hartmann M, Gebert S, Jarschel M, Zinner T, Tran-Gia P (2013) Pareto-optimal resilient controller placement in SDN-based core networks. In: Proceedings of the 2013 25th international teletraffic congress (ITC), pp 1-9. https://doi.org/10.1109/ITC.2013.6662939
6. Tanha M, Sajjadi D, Ruby R, Pan J (2018) Capacity-aware and delay-guaranteed resilient controller placement for software-defined WANs. IEEE Trans Netw Serv Manag 15(3):991–1005. https://doi.org/10.1109/TNSM.2018.2829661
7. Vizarreta P, Machuca CM, Kellerer W (2016) Controller placement strategies for a resilient SDN control plane. In: 2016 8th international workshop on resilient networks design and modeling (RNDM), pp 253–259. https://doi.org/10.1109/RNDM.2016.7608295
8. Moazzeni S, Khayyambashi MR, Movahhedinia N, Callegati F (2018) On reliability improvement of software-defined networks. Comput Netw 133:195–211. ISSN 1389-1286. https://doi.org/10.1016/j.comnet.2018.01.023
9. Singh AK, Maurya S, Kumar N, Srivastava S (2020) Heuristic approaches for the reliable SDN controller placement problem. Trans Emerg Telecommun Technol 31(2):e3761
10. Jalili A, Keshtgari M, Akbari R (2020) A new framework for reliable control placement in software-defined networks based on multi-criteria clustering approach. Soft Comput 24(4):2897–2916

11. Mohan PM, Truong-Huu T, Gurusamy M (2020) Byzantine-resilient controller mapping and remapping in software defined networks. IEEE Trans Netw Sci Eng 7(4):2714–2729. https://doi.org/10.1109/TNSE.2020.2981521
12. Petale S, Thangaraj J (2019) Failure-based controller placement in software defined networks. IEEE Trans Netw Serv Manag 17(1):503–516
13. Aravind P, Saradhi Varma GP, Prasad Reddy PVGD (2021) Simulated annealing based optimal controller placement in software defined networks with capacity constraint and failure awareness. J King Saud Univ Comput Inf Sci. ISSN 1319-1578. https://doi.org/10.1016/j.jksuci.2021.04.012
14. Mirjalili S (2015) Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm. Knowl-Based Syst 89:228–249
15. http://www.topology-zoo.org/dataset.html

Smart Weather Prediction Using Machine Learning

Suvendra Kumar Jayasingh, Jibendu Kumar Mantri, and Sipali Pradhan

Abstract Prediction of weather is a challenging task for all researchers of weather and the meteorological department. Many techniques are evolved in time for the prediction of weather since last many years. The advancement of science and technology has helped the researchers to perform the prediction of weather simply and with less error rates. The soft computing techniques are the new technologies in computer science which are capable of making the weather prediction with promising output and less error rates. Weather prediction is done traditionally by use of many historical data in many models of physics. This prediction of weather is unsteady due to the change in weather condition. Because of change of weather system, the prediction is unstable. In this paper, we present different machine learning models that will make use of the historical data to train the models and then the model will be used to predict the weather whose accuracy is better than the traditional models. The evaluation of the models on the basis of accuracy shows that the models outperform and can be used as state-of-art technique to predict the weather in smarter way in less time.

Keywords Random forest · Decision tree · Support vector machine · Gradient boosting

1 Introduction

Climate modelling and weather prediction is the application of science and technology to predict the state of the atmosphere for a given location is a challenging task for the researchers in this modern age. The use of machine learning for prediction of weather uses the dataset of 21 years having the parameters temp, dew, humidity, pressure, visibility and windspeed. The event of the place will be predicted by use of the models. The events may be "No Rain", "Fog", "Rain, Thunderstorm", "Thunderstorm", "Fog, Rain", etc. The machine learning models under consideration in

S. K. Jayasingh · J. K. Mantri (✉) · S. Pradhan
Department of Computer Application, Maharaja Sriram Chandra Bhanja Deo University, Baripada, Odisha, India
e-mail: jkmantri@gmail.com

this paper are random forest, decision tree, support vector machine, KNN, Adaboost, Xgboost, Gradient Boosting, naïve Bayes and logistic regression, etc. The evaluation of all these models are done on the basis of their performance compared as per their accuracy and f1 score.

2 Literature Review

Singh et al. [1] have made the weather forecasting using machine learning algorithms. They used different machine learning algorithms to predict the weather events. Khajure and Mohod [2] have deployed the future weather forecasting using soft computing techniques. Bhardwaj and Duhoon [3] have made use of soft computing techniques for forecasting of weather. Haghbin et al. [4] applied soft computing models for predicting sea surface temperature and made the review and assessment. Vathsala and Koolagudi [5] have used neuro-fuzzy model for quantified rainfall prediction using data mining and soft computing approaches. Balogh et al. [6] made a toy model investigate stability of AI-based dynamical systems. Jayasingh et al. [7] have shown a novel approach for data classification using neural network. Litta et al. [8] have used artificial neural network model in prediction of meteorological parameters during premonsoon thunderstorms. Schultz et al. [9] have shown if deep learning beat numerical weather prediction. Sharma and Agarwal [10] have explained temperature prediction using wavelet neural network. Lin et al. [11] have discussed time series prediction based on support vector regression. Askari and Askari [12] have used time series grey system prediction-based models for gold price forecasting. Lee and Lee [13] have constructed efficient regional hazardous weather prediction models through big data analysis. Jayasingh et al. [14] have made weather prediction using hybrid soft computing models. Sofian et al. [15] have done monthly rainfall prediction based on artificial neural networks with back propagation and radial basis function.

3 Methodology

In our research, we are trying to compare the different machine learning techniques for predicting the events. Machine learning models are designed to predict the events on the basis of the temp, dew, humidity, pressure, visibility and windspeed. In this research, we have taken the weather data from 1996 to 2017 to train our machine learning model. Here, we are using random forest, deceison tree, support vector regression, KNN, Adaboost, gradian-boost, Xgboost, naïve Bayes and logistic regression to evaluate and train our model. The flow diagram of our proposed methodology is shown in Fig. 1.
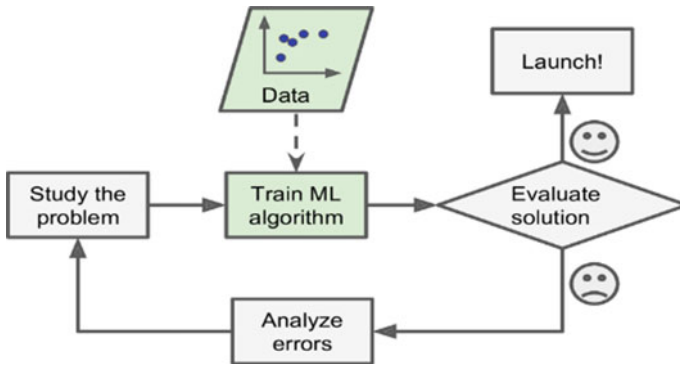
Fig. 1 The flow diagram of our proposed methodology

Initially, we preprocessed the dataset by eliminating any incomplete record. We apply feature selection and eliminate features that have no direct impact on the performance. Then we have to encode the categorical features into numerical feature because our machine learning algorithms will not be able to understand the categorical value. Here when we try to encode the categorical feature into numerical using label encoder, our model not showed a good accuracy score that's why here we used get dummies method to encode the categorical features. We also checked that whether we have outliers or not but we did not find any outliers. And also we took a look at the multicollinearity and drop some highly co-related columns because if we keep some highly correlated features means it will affect our model accuracy and then we did some feature scaling.

Once our dataset is split using train test split on 80:20 ratio, the predictive models like linear regression, ridge and lasso regression, random forest, decision tree, support vector regressor. Adaboost, gradiantboost and xgboost are used to forecast upcoming match results. The machine learning process is shown in Fig. 2.
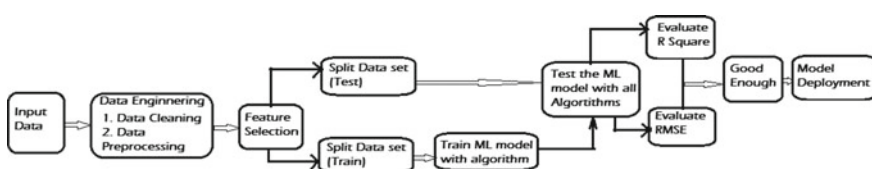
Fig. 2 Machine learning process overview

4 Techniques Used

K-nearest neighbours (KNN)

K-nearest neighbours (KNN) is a supervised learning method that may be used for both regression and classification, however it is usually utilized for classification. KNN aims to predict the right class of testing data given a set with various classes by calculating the distance between both the testing data and all the training points. It then chooses the k points that are the most similar to the test.

After the points have been chosen, the algorithm calculates the likelihood (in the classification phase) that the test point belongs to one of the k training point classes, and the class with the greatest probability is chosen. The regression model in a regression issue is the mean of the k chosen training points data as shown in Fig. 3.

Lazy learners are a phrase used to describe KNN algorithms. Eager learners refer the techniques such as Bayesian classification, logistic regression, SVM and others. These methods generalize over the training set before getting the test data that is the model is trained using training dataset and then test data is received by the model, and then predict/classify the test data.

With the KNN method, however, this is not the case. For the training set, it does not construct a generalized model; instead, it waits for the test data. Only when test data has been supplied it begins generalizing the training data in order to categorize
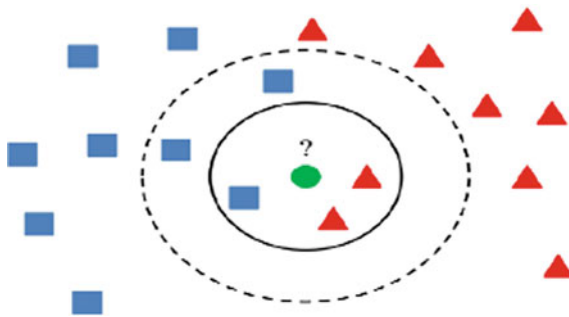
Fig. 3 K-nearest neighbours

the test data. As a result, a lazy learner just saves the training data and waits for the test set to arrive. Such algorithms are more effective at categorizing a particular test dataset than they are during training.

```
generate_pred_knn("KNN",KNN,x_train,x_test,y_train,y_test)

------Evaluation metrics for training data set--------
modelname- KNN
rmse is 8.889448004262103
Rsqr is  91.24
-------Evaluation metrics for test dataset--------
modelname- KNN
rmse is 12.324286094952836
Rsqr is  83.08
```

Support Vector Machine

By the computer hardware advancements, as well as the increasing availability and low cost of technology elements such as RAM and GPU, a substantial chunk of labelled data is readily available and created on a regular basis. To make the firm less sensitive to unanticipated situations, we focus on predictive data processing. The support vectir machine as shown in Fig. 4 is used for the pupose of classification.

```
generate_pred_svr("SVR",svr,x_train,x_test,y_train,y_test)

------Evaluation metrics for training data set--------
modelname- SVR
rmse is 27.551063062094208
Rsqr is  15.82
-------Evaluation metrics for test dataset--------
modelname- SVR
rmse is 27.576245165975607
Rsqr is  15.3
```
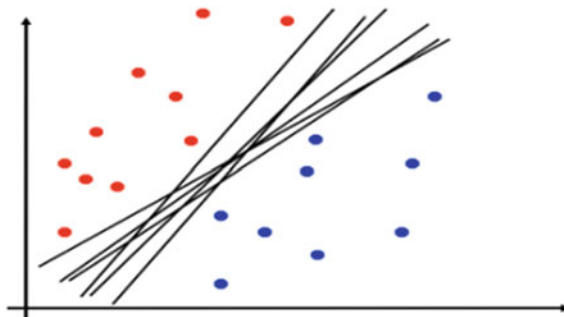

Fig. 4 Support vector machine

Decision Tree

A decision tree accepts an item or scenario with a set of attributes as input and provides a yes/no "decision". It is one of the best used effective techniques of machine learning. We explain how to acquire a good hypothesis by first describing the depiction hypothesis space.

```
generate_pred_dt("DT",DT,x_train,x_test,y_train,y_test)

------Evaluation metrics for training data set--------
modelname- DT
rmse is 0.0
Rsqr is  100.0
-------Evaluation metrics for test dataset--------
modelname- DT
rmse is 6.773538410360594
Rsqr is  94.89
```

Each node examines the value of one or more input attributes. Leaf nodes give the values to be delivered if that leaf is encountered. Branches from the node correspond to possible attribute values in the **Decision Tree.**

Random Forests

It is a collection of decision trees that have been bagged trained as shown in Fig. 5. The strategy generates more tree variety, which trades a higher bias for fewer switches,
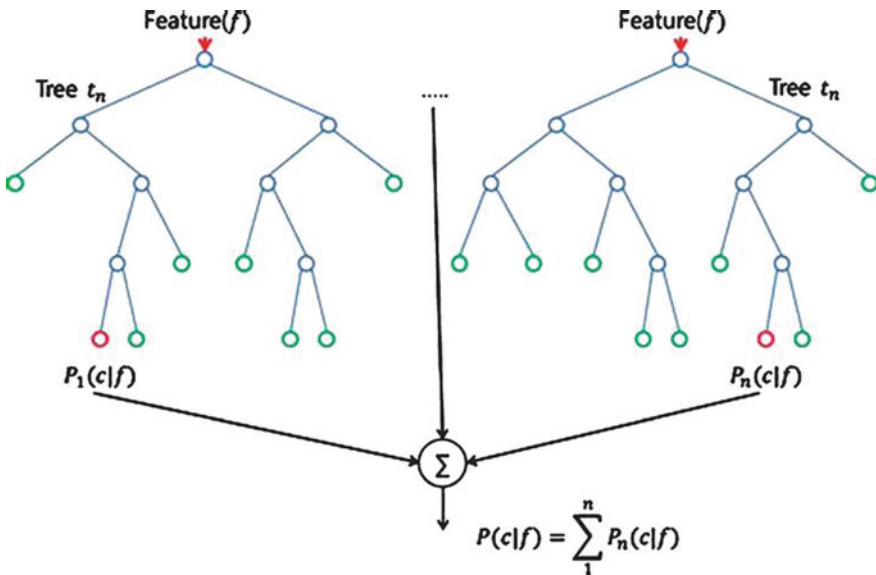
Fig. 5 Random forest

resulting in a more robust overall model. The random forest regression that comes after it is similar to the bagging regression that came before it.

```
generate_pred_rf("RF",RF,x_train,x_test,y_train,y_test)

------Evaluation metrics for training data set--------
modelname- RF
rmse is 1.4639971789356376
Rsqr is  99.76
-------Evaluation metrics for test dataset-------
modelname- RF
rmse is 3.766502524372585
Rsqr is  98.42
0.9841982292297486
```

XGboost

It stands for "Extreme Gradient Boosting". Boost is a toolkit for distributed gradient boosting that has been optimized for speed, versatility and portability. Machine learning algorithms are built using the Gradient Boosting framework as shown in Fig. 6. It employs parallel tree boosting to quickly and accurately solve a number of data science problems.
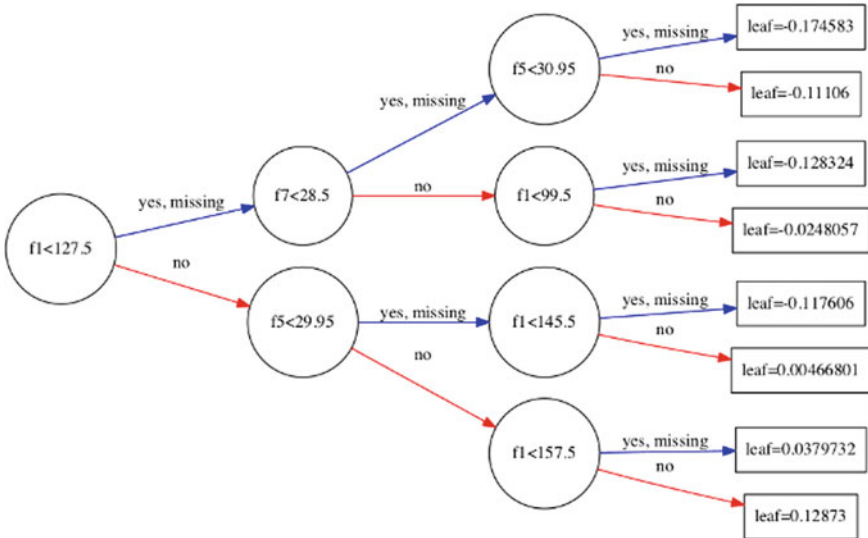
Fig. 6 Xgboost
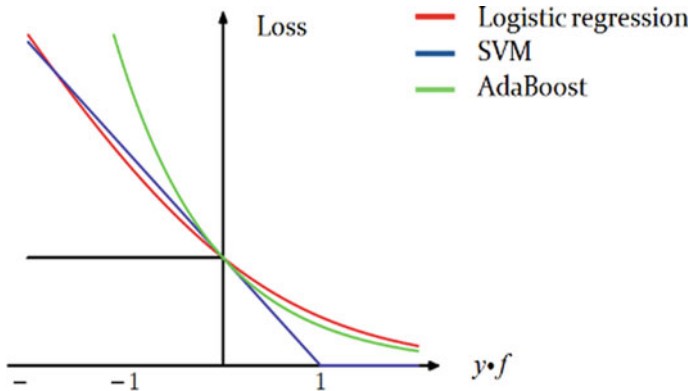
Fig. 7 Adaboost

```
generate_pred_xg("xgboost",xg,x_train,x_test,y_train,y_test)

------Evaluation metrics for training data set--------
modelname- xgboost
rmse is 5.043328069820755
Rsqr is  97.18
-------Evaluation metrics for test dataset--------
modelname- xgboost
rmse is 5.526413482015406
Rsqr is  96.6
```

Adaboost

Showing genuine Freund and Robert Schapire developed Adaboost (Adaptive Boosting), a statistical categorization method for which they were given the Mathematicians Award in 2003. It can be combined with a number of learning algorithms to improve results. The output of other learning algorithms ("learning and self") is combined into a weighted sum that reflects the proper outcome of the boosted classifier as shown in Fig. 7.

```
generate_pred_ada("Adaboost",ada,x_train,x_test,y_train,y_test)

------Evaluation metrics for training data set--------
modelname- Adaboost
rmse is 23.491058448773366
Rsqr is  38.8
-------Evaluation metrics for test dataset--------
modelname- Adaboost
rmse is 23.502830496434207
Rsqr is  38.47
```

Gradient Boosting

A prominent boosting technique is gradient boosting. Each predictor in gradient boosting corrects the mistake of its preceding as shown in Fig. 8. With exception of

Understanding the Algorithm
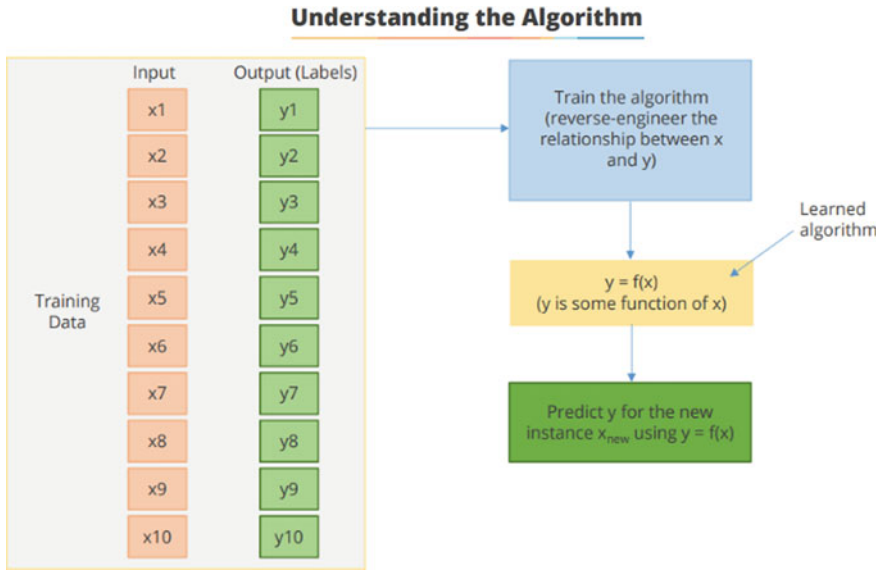
Fig. 8 Gradient boosting

Adaboost, the training instance parameters are not adjusted; instead, each predictor is trained using the immediate predecessor residual as labels.

CART is the foundation learner in a method called Gradient Boosted Trees (Classification and Regression Trees).

```
generate_pred_gdb("Gradiantboosting",gbr,x_train,x_test,y_train,y_test)

------Evaluation metrics for training data set--------
modelname- Gradiantboosting
rmse is 18.284885131258374
Rsqr is  62.92
-------Evaluation metrics for test dataset--------
modelname- Gradiantboosting
rmse is 18.369615400396928
Rsqr is  62.41
```

5   Data Analysis

Features description

The weather data is collected for prediction analysis for 21 years (1996–2017) and stored in a dataset. There are 7 variables and 7750 instances in the dataset used as shown in Table 1.

Table 1 Dataset used in the experiment

Temp		Dew	Humidity	Pressure	Visibility	Wind	Event
0	28	24	76	1002	5	11	No rain
1	29	26	85	1003	5	8	No rain
2	32	26	78	1004	5	11	No rain
3	31	26	81	1003	4	13	No rain
4	31	26	86	1001	4	10	Rain, thunderstorm

Table 2 Accuracy of different models

S. No.	Model name	Accuracy
1	Random forest	79.52
2	Decision tree	71.23
3	Support vector machine	59.33
4	KNN	77.86
5	Adaboost	71.43
6	Xgboost	79.94
7	Gradient boosting	81.67
8	Naïve Bayes	73.09
9	Logistic regression	78.14

6   Result Analysis

The different machine learning models were trained with the weather data of 21 years and the model was used to predict the test data. Hence, the performance of all the models based on their accuracy is summarized in Table 2. The graphical comparison of accuracy of these models is shown in Fig. 9.

The different machine learning models were trained with the weather data of 21 years and the model was used to predict the test data. Hence, the performance of all the models based on their $f1$ score is summarized in Table 3 and the graphical comparison of $f1$ score of these models is shown in Fig. 10.

7   Conclusions and Future Work

We can estimate the weather events using a machine learning model that takes into account the different weather parameters. In this paper, we presented different machine learning models which can be used for prediction of weather with much simpler and easier way than the physical models. The accuracy evaluation of the models shows that the machine learning models perform better than the traditional
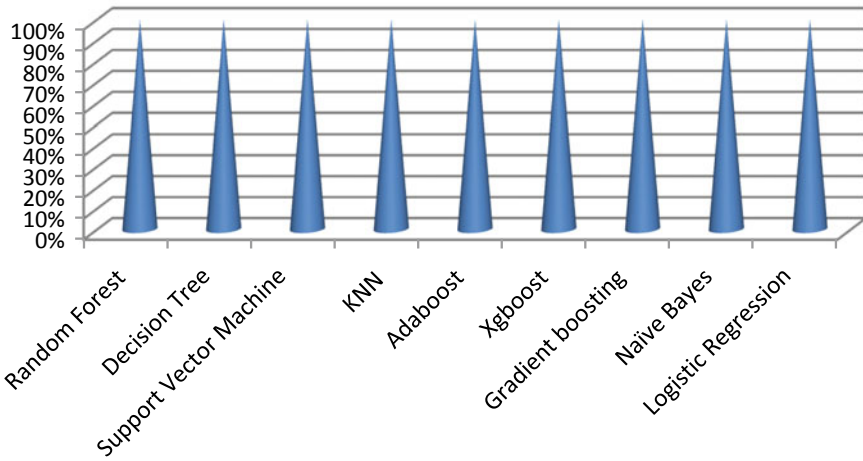
Fig. 9 The graphical comparison of accuracy of different models

Table 3 $F1$ score of different models

S. No.	Model name	$F1$ score
1	Random forest	0.81
2	Decision tree	0.75
3	Support vector machine	0.74
4	KNN	0.80
5	Adaboost	0.74
6	Xgboost	0.80
7	Gradient boosting	0.83
8	Naïve Bayes	0.79
9	Logistic regression	0.80

models. These models made use of the dataset collected from predefined recourses in which the maximum accuracy is observed upto 81.67%. In future, is is planned to use the different IoT devices to collect the accurate data so that the data set to be used in the model will be more exact and accordingly the performance of the model will be more correct.
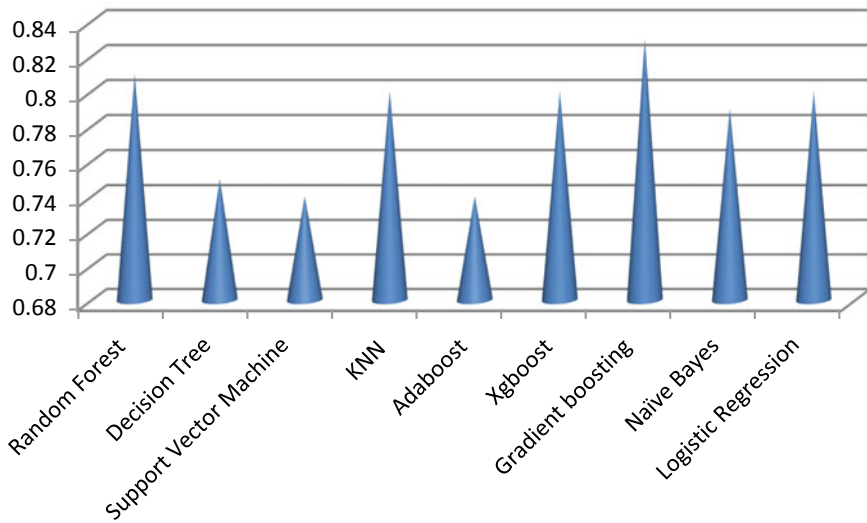
Fig. 10 The graphical comparison of $F1$ score of different models

References

1. Singh N, Chaturvedi S, Akhter S (2019) Weather forecasting using machine learning algorithm. In: International conference on signal processing and communication (ICSC), pp 171–174. https://doi.org/10.1109/ICSC45622.2019.8938211

2. Khajure S, Mohod SW (2016) Future weather forecasting using soft computing techniques. Proc Comput Sci 78:402–407. ISSN 1877-0509. https://doi.org/10.1016/j.procs.2016.02.081

3. Bhardwaj R, Duhoon V (2018) Weather forecasting using soft computing techniques. In: International conference on computing, power and communication technologies (GUCON), pp 1111–1115. https://doi.org/10.1109/GUCON.2018.8675088

4. Haghbin M, Sharafati A, Motta D (2021) Applications of soft computing models for predicting sea surface temperature: a comprehensive review and assessment. Prog Earth Planet Sci 8(4). https://doi.org/10.1186/s40645-020-00400-9

5. Vathsala H, Koolagudi SG (2021) Neuro-fuzzy model for quantified rainfall prediction using data mining and soft computing approaches. IETE J Res. https://doi.org/10.1080/03772063. 2021.1912648

6. Balogh B, Saint-Martin D, Ribes A (2021) A toy model to investigate stability of AI-based dynamical systems. Geophys Res Lett 48(8). https://doi.org/10.1029/2020GL092133

7. Jayasingh SK, Gountia D, Samal N, Chinara PK (2021) A novel approach for data classification using neural network. IETE J Res. https://doi.org/10.1080/03772063.2021.1986152

8. Litta AJ, Idicula SM, Mohanty UC (2013) Artificial neural network model in prediction of meteorological parameters during premonsoon thunderstorms. Int J Atmos Sci 2013:14. Article ID 525383. https://doi.org/10.1155/2013/525383

9. Schultz MG, Betancourt C, Gong B, Kleinert F, Langguth M, Leufen LH, Mozaffari A, Stadtler S (2021) Can deep learning beat numerical weather prediction. Phil Trans R Soc A 379:20200097. https://doi.org/10.1098/rsta.2020.0097

10. Sharma A, Agarwal S (2012) Temperature prediction using wavelet neural network. Res J Inf Technol 4:22–30. https://doi.org/10.3923/rjit.2012.22.30. https://scialert.net/abstract/?doi=rjit. 2012.22.30

11. Lin S, Wang G, Zhang S, Li J (2006) Time series prediction based on support vector regression. Inf Technol J 5:353–357. https://doi.org/10.3923/itj.2006.353.357. https://scialert.net/abstract/?doi=itj.2006.353.357

12. Askari M, Askari H (2011) Time series grey system prediction-based models: gold price forecasting. Trends Appl Sci Res 6:1287–1292. https://doi.org/10.3923/tasr.2011.1287.1292. https://scialert.net/abstract/?doi=tasr.2011.1287.1292

13. Lee J, Lee J (2016) Constructing efficient regional hazardous weather prediction models through big data analysis. IJFIS 16:1–12. https://doi.org/10.5391/IJFIS.2016.16.1.1

14. Jayasingh SK, Mantri JK, Pradhan S (2021) Weather prediction using hybrid soft computing models. In: Udgata SK, Sethi S, Srirama SN (eds) Intelligent systems. Lecture notes in networks and systems, vol 185. Springer, Singapore. https://doi.org/10.1007/978-981-33-6081-5_4

15. Sofian IM, Affandi AK, Iskandar I, Apriani Y (2018) Monthly rainfall prediction based on artificial neural networks with backpropagation and radial basis function. Int J Adv Intell Inf 4(2):154–166. ISSN 2548-3161. https://doi.org/10.26555/ijain.v4i2.208

Flood Susceptibility Modeling Using Forest-Based Regression

Bibhu Prasad Mishra, Dillip Kumar Ghose, Deba Prakash Satapathy, and Sourav Ghose

Abstract Flood is a disastrous event, having significant impact on environment and human life. Flood susceptibility modeling decreases the damage caused by floods, which is a chief component of water management. This study employs a forest-based classification and regression (FBR) in flood susceptibility analysis. A flood susceptibility map was developed using the flood influencing factors and a flood inventory map using FBR. ROC curve was utilized to study the efficiency of the model, having AUC value of 0.997, R-squared value 0.98 during training and 0.92 during validation.

1 Introduction

The harmful effect of floods significantly increases in last decades [1, 2] which has a very high impact on the lifestyle of people, leads to a huge damage, and loss of life [3–6]. Flood causes severe natural hazards whose genesis is related to both artificial and natural factors [7–10]. Forecasting helps to mitigate the damages caused by the floods. GIS combined with various machine learning (ML) algorithms to provide a very accurate process to delineate areas prone to flood occurrence [1, 11, 12, 13]. To increase the accuracy of the final results, ML models lead to flood susceptibility mapping, by using the algorithm like: k-nearest neighbor [6], artificial neural network

B. P. Mishra · D. P. Satapathy
Department of Civil Engineering, Odisha University of Technology and Research, Bhubaneswar, Odisha, India
e-mail: dpsatapathy@cet.edu.in

D. K. Ghose (✉)
Department of Civil Engineering, National Institute of Technology Silchar, Assam, India
e-mail: dillip@civil.nits.ac.in

S. Ghose
Department of Civil Engineering, Veer Surendra Sai University of Technology, Burla, Odisha, India

[14], logistic regression [15], Naïve Bayes [16], support vector machine [17], neuro-fuzzy inference system [18], random forest [19], and deep learning neural network [1]. Ensemble models involve many algorithms to predict an accurate evaluation of the flash flood susceptibility for a given territory [11].

The forest-based regression (FBR) is statistical model [20]. It was widely used by the researchers around the globe for its efficient performance. FBR is mainly used for extraction of several samples from dataset to create a decision tree for every sample in bootstrapping. The prediction of decision trees is combined to obtain the result. Idea of FBR is to associate various weak decision trees for producing a robust decision tree. The efficacy of the decision was improved with reference to weak decision tree. FBR model has highest stability in context to artificial neural network (ANN), and it is not prone to overfitting [18, 21].

The objective of the present study is to develop a FBR model at Basantpur watershed of Mahanadi River and to evaluate the performance of the model for developing flood susceptibility map.

2   Study Area

The Basantpur watershed of Mahanadi River basin is located in the state of Chhattisgarh, India between $20°20'5''$ to $22°45'20''$ north and $80°24'35''$ to $82°40'51''$ east. This watershed is an important area for industrialization and agriculture, creating environmental and commercial benefits for Chhattisgarh and Odisha state governments and the local people. The area of the watershed is 31,165 km^2.

3   Materials and Methodology

3.1   Data Preparation in GIS

In this paper, ArcMap 10.8 is used to demarcate the Basantpur watershed. Topographic factors such as altitude, aspect, curvature, SPI, STI, slope, TRI, TWI, and streams characteristics are extracted from the digital elevation model (DEM) which is presented in Fig. 2. DEM with resolution 2.5 m × 2.5 m was collected from geoplatform of the Indian Space Research Organization (ISRO). Distance from the river was calculated by using the Euclidian distance tools of ArcMap shown in Fig. 2. Decadal LULC is used to prepare LULC map as shown in Fig. 1. NDVI, lithology, and soil maps are collected from National Information System for Climate and Environmental Studies (NICES) shown in Fig. 2. Drainage density map was prepared by using the density tools in ArcMap. A flood inventory map was prepared by the last fifty years flood events. To construct a flood susceptibility map, parameter sets related to floods are required [22].
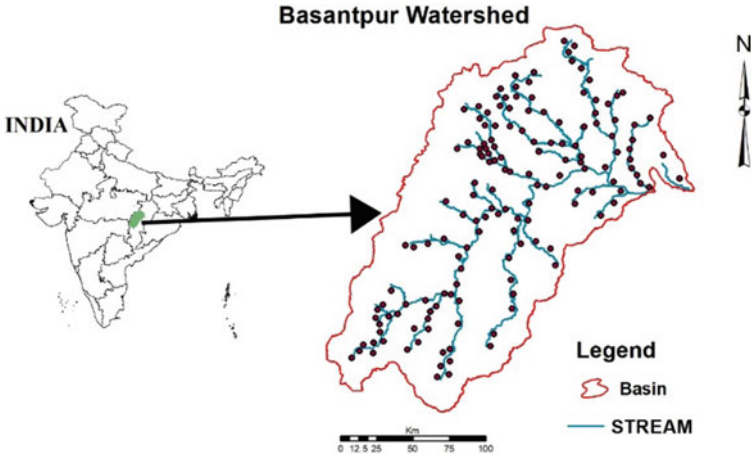
Fig. 1 Study area and flood inventory

3.2 Forest-Based Regression

FBR comprises self-determining regression decision trees $\{h(x, \theta_k), \quad k = 1, 2, 3, \ldots N\}$ where θ_k represents a unique distributed random variable, where x is input and N = number of regression decision trees and k = mean value of regression trees, where $h(x, \theta_t)$ is predicted result [23]. ArcGIS Pro 10.8 is used to obtain the result. The procedure is

(1) Bootstrap method randomly extracts original data to generate k sets, i.e., $\theta_1, \theta_2, \theta_3, \ldots, \theta_k$; and k regression trees of, $h(x, \theta_1), h(x, \theta_2), h(x, \theta_3) \ldots, h(x, \theta_k)$.
(2) Principle of every regression tree brings restriction without splitting.
(3) Random forest consists of k regression trees which is created by above process.

During the process of forest generation, probability of each sample is not extracted. 40% of the total sample was out of bag (OOB) data. Model accuracy is evaluated by calculating the OOB error without bias. Mean squared error for the FBR model is used to evaluate factor of influence for the target variable, and importance of each variable is measured [23, 24]. The precise execution is to evaluate the root mean square error value of the original OOB dataset, i.e., RMSE_{OOB}, to replace the target variable x_i via OOB data $\text{RMSE}_{\text{OOBi}}$. Then, z-score standardization for the importance of target variable x_i is obtained, which is mathematically expressed as,

$$\text{Importance of variable} = \left[\frac{1}{U} \sum_{j=1}^{u} (\text{MSE}_{\text{oob}} - \text{MSE}_{\text{oobi}}) \right] / \sigma_{\text{SD}}, \quad (1 \leq i \leq K)$$
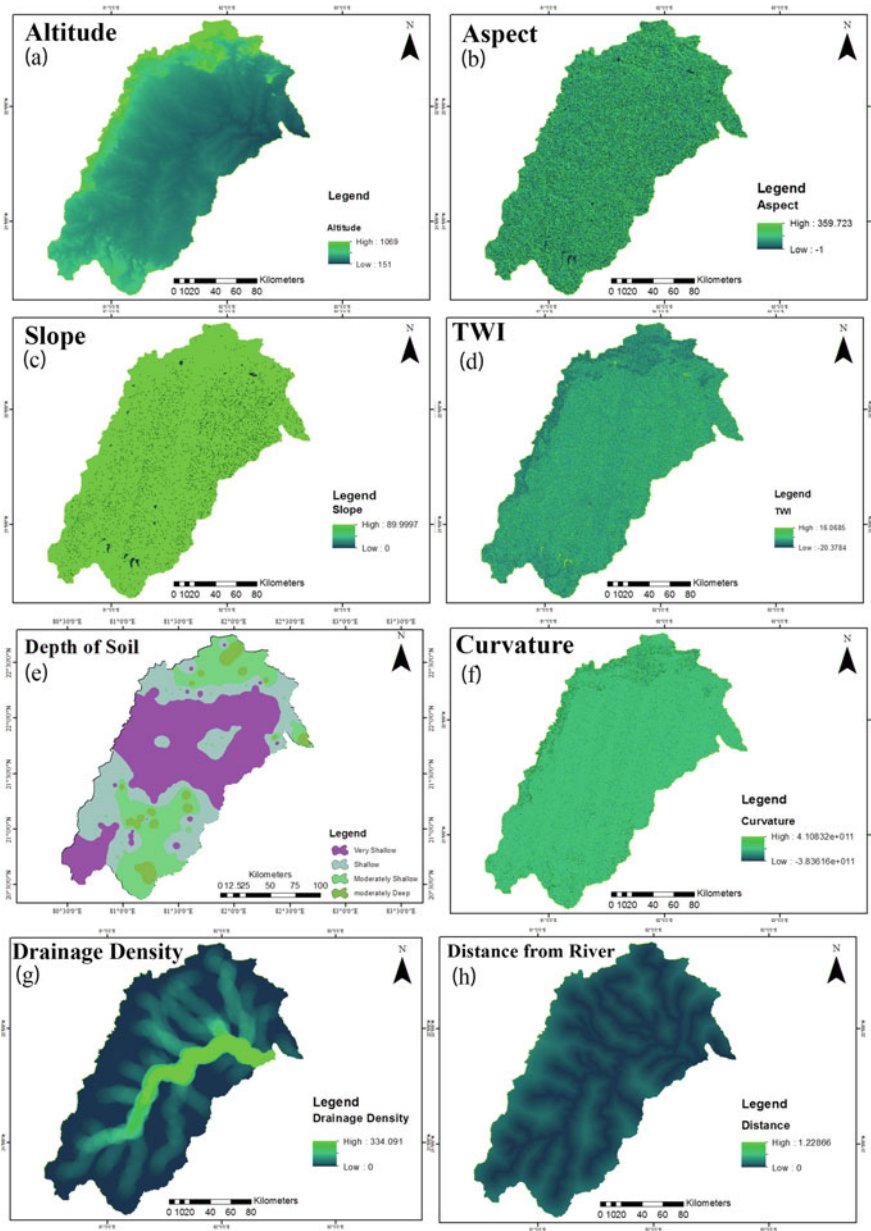
Fig. 2 Flood influencing factors
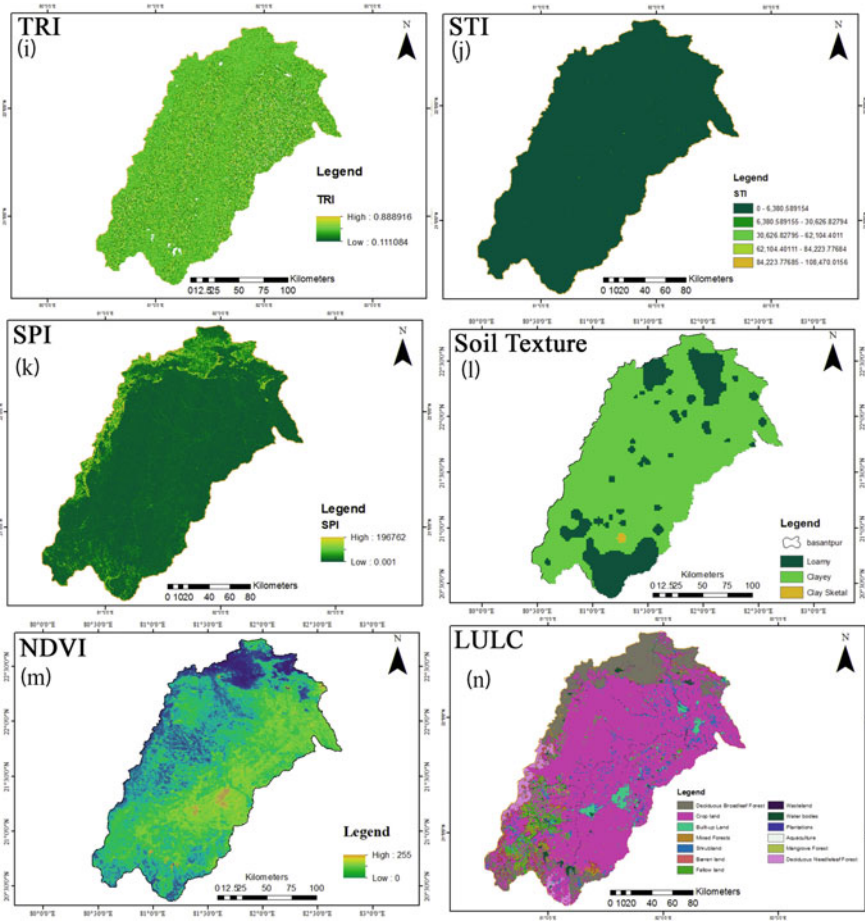
Fig. 2 (continued)

3.3 Evaluation of Model

Before construction of model, all data were randomly divided into two distinctive parts, training data and validation data which includes 70% and 30% of total data, respectively, for model evaluation. Model performance was evaluated using tenfold cross-validation to evaluate the ability of the model on new data. This is a resampling procedure where all data are randomly divided into ten folds; at each run of the given model, one fold is held out for validation, and the remaining $k-1$ folds are used for training [25]. The average accuracy of all folds is conveyed as the final accuracy. The validation index in both training and validation phases was R-squared and mean squared error which measures the ability of the calibrated model to decide flood and non-flood events. Area under the receiver operating characteristic (ROC) curve

(AUROC) was used to validate the model performance. For superior performance of the model, value of RMSE must be smaller, whereas R-squared and AUROC must be higher.

4 Results and Discussions

A flood susceptibility maps were generated using FBR, Fig. 3. The range of flood susceptible map was range from 0 to 1, 0 indicating lowest, and 1 indicates highest flood. The obtained values are divided into four categories that is very low, low, high, and very high. The watershed was divided into these categories, and the flood prone areas are shown in Table 1.
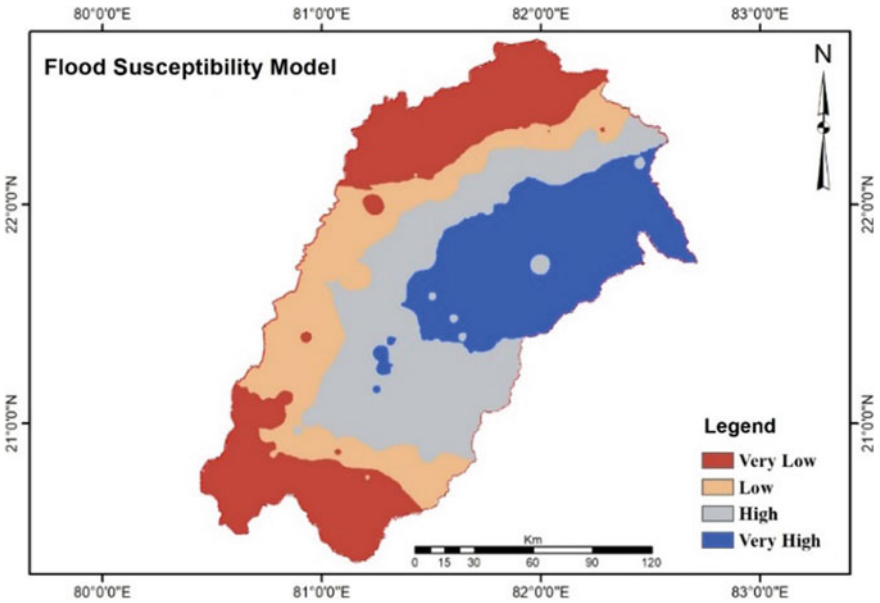
Fig. 3 Flood susceptibility map

Table 1 Flood-prone areas in percentage

Susceptible area	FBR
Very low	11.77
Low	15.41
Moderate	17.31
High	55.51
Very high	11.77

Table 2 Performance index of FBR

Criteria of evaluation	Training	Validation	Difference (δ)
R-square	0.98	0.92	0.06
RMSE	0.06	0.09	0.03

In present research, a simple and efficient model is proposed for predicting flood susceptibility, and features of input datasets are also considered since it affects the performance of the models. The efficacy of the model was evaluated through R-square and RMSE indices. Results of the training and the validation set are shown in Table 2.

As presented in Table 2, R-square for training and validation datasets is 0.98 and 0.92, respectively. The RMSE values for training and validation set are found to be 0.06 and 0.09, correspondingly. Along with the small value of δ, difference between training and validation indicates simplification ability, and results model has high prediction accuracy and outstanding capability which can be used to mitigate the flood susceptibility. The obtained results were validated with the help of the ROC plot Fig. 4. Both the success rate and the prediction rate were constructed. The most significant statistical measure of ROC curve is represented by the area under the
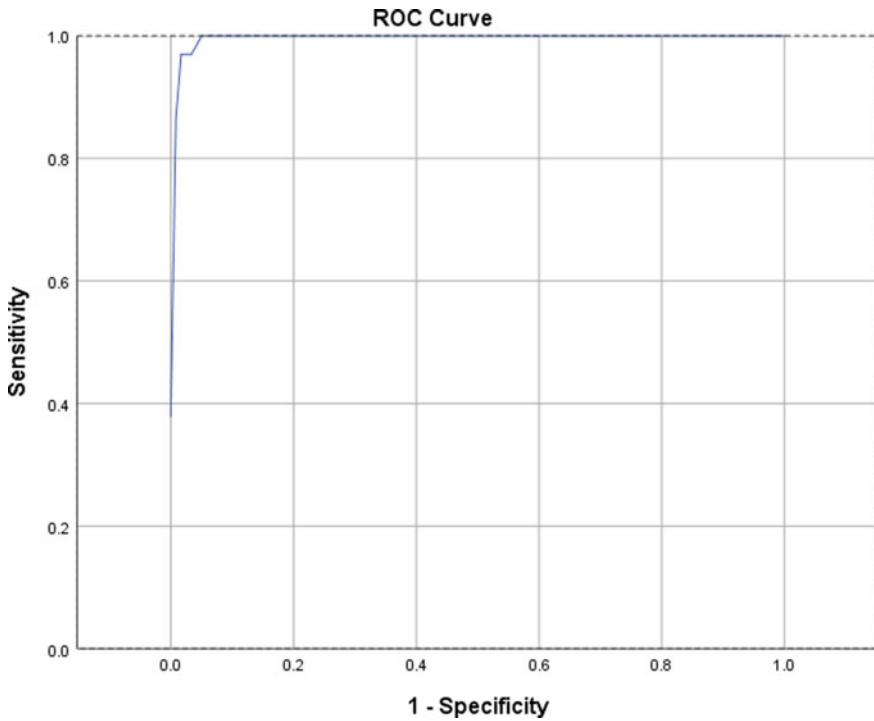
Fig. 4 AUROC curve of the model performance

curve (AUC). It is notable that the success rate was used to validate the results, while the prediction rate, designed using the validating sample, highlights the accuracy of the results. Regarding prediction rate, the AUC value obtained was 0.997.

5 Conclusion and Future Scope

The current study was proposed to evaluate FBR method for flood susceptibility mapping in the Basantpur watershed of Mahanadi River basin, India. The obtained FBR model is verified through the model evaluation criteria and holds an outstanding performance based on AUROC and R-square. AUROC for the model is 0.997, and R-square for training and testing phase is 0.98 and 0.92, respectively. Influential factors for flood accuracy can increase the accuracy of the generated susceptibility maps. Future study would include exploring the influencing factors based on the dependencies for the input to include developing deep ANN models which can be utilized to predict the susceptibility.

References

1. Bui DT, Tsangaratos P, Ngo P-TT, Pham TD, Pham BT (2019) Flash flood susceptibility modeling using an optimized fuzzy rule-based feature selection technique and tree-based ensemble methods. Sci Total Environ 668:1038–1054
2. Sahoo A, Ghose DK (2021) Flood frequency analysis for menace gauging station of Mahanadi river, India. J Inst Eng (India) Ser A 1–12
3. Janizadeh S, Avand M, Jaafari A, Phong TV, Bayat M, Ahmadisharaf E, Prakash I, Pham BT, Lee S (2019) Prediction success of machine learning methods for flash flood susceptibility mapping in the tafresh watershed, Iran. Sustain 11:5426
4. Samantaray S, Sahoo A, Agnihotri A (2021) Assessment of flood frequency using statistical and hybrid neural network method: Mahanadi river basin, India. J Geol Soc India 97(8):867–880
5. Sahoo A, Samantaray S, Paul S (2021) Efficacy of ANFIS-GOA technique in flood prediction: a case study of Mahanadi river basin in India. H2Open J 4(1):137–156
6. Sahoo A, Singh UK, Kumar MH, Samantaray S (2021) Estimation of flood in a river basin through neural networks: a case study. In: Communication software and networks, pp 755–763. Springer, Singapore
7. Youssef AM, Sefry SA, Pradhan B, Alfadail EA (2016) Analysis on causes of flash flood in Jeddah city (Kingdom of Saudi Arabia) of 2009 and 2011 using multi-sensor remote sensing data and GIS. Geomat Nat Haz Risk 7:1018–1042
8. Sahoo A, Samantaray S, Ghose DK (2021) Prediction of flood in Barak river using hybrid machine learning approaches: a case study. J Geol Soc India 97(2):186–198
9. Samantaray S, Sahoo A (2021) Estimation of flood frequency using statistical method: Mahanadi river basin, India. H2Open J 3(1):189–207
10. Sahoo A, Samantaray S, Bankuru S, Ghose DK (2020) Prediction of flood using adaptive neuro-fuzzy inference systems: a case study. In: Smart intelligent computing and applications, pp 733–739. Springer, Singapore
11. Arabameri A, Saha S, Chen W, Roy J, Pradhan B, Bui DT (2020) Flash flood susceptibility modelling using functional tree and hybrid ensemble techniques. J Hydrol 125007

12. Cao C, Xu P, Wang Y, Chen J, Zheng L, Niu C (2016) Flash flood hazard susceptibility mapping using frequency ratio and statistical index methods in coalmine subsidence areas. Sustainability 8:948

13. Costache R, Hong H, Pham QB (2020) Comparative assessment of the flash-flood potential within small mountain catchments using bivariate statistics and their novel hybrid integration with machine learning models. Sci Total Environ 711:134514

14. Chapi K, Singh VP, Shirzadi A, Shahabi H, Bui DT, Pham BT, Khosravi K (2017) A novel hybrid artificial intelligence approach for flood susceptibility assessment. Environ Model Softw 95:229–245

15. Ali SA, Parvin F, Pham QB, Vojtek M, Vojteková J, Costache R, Linh NTT, Nguyen HQ, Ahmad A, Ghorbani MA (2020) GIS-based comparative assessment of flood susceptibility mapping using hybrid multi-criteria decision-making approach, naïve Bayes tree, bivariate statistics and logistic regression: a case of Topľa basin, Slovakia. Ecol Indic 117:106620

16. Chen W, Li Y, Xue W, Shahabi H, Li S, Hong H, Wang X, Bian H, Zhang S, Pradhan B (2020) Modeling flood susceptibility using data-driven approaches of naïve bayes tree, alternating decision tree, and random forest methods. Sci Total Environ 701:134979

17. Choubin B, Moradi E, Golshan M, Adamowski J, Sajedi-Hosseini F, Mosavi A (2019) An Ensemble prediction of flood susceptibility using multivariate discriminant analysis, classification and regression trees, and support vector machines. Sci Total Environ 651:2087–2096

18. Wang Y, Hong H, Chen W, Li S, Panahi M, Khosravi K, Shirzadi A, Shahabi H, Panahi S, Costache R (2019) Flood susceptibility mapping in Dingnan county (China) using adaptive neuro-fuzzy inference system with biogeography-based optimization and imperialistic competitive algorithm. J Environ Manage 247:712–729

19. Lee S, Kim J-C, Jung H-S, Lee MJ, Lee S (2017) Spatial prediction of flood susceptibility using random-forest and boosted-tree models in Seoul metropolitan city, Korea. Geomat Nat Haz Risk 8:1185–1203

20. Breiman L (2001) Random forests. Mach Learn 45:5–32

21. Liu XP, Li X, Ye JA et al. (2007) Using ant colony intelligence to mine the conversion rules of geographic cellular automata. China Sci 2007(06):824–834

22. Tehrany MS, Pradhan B, Jebur MN (2014) Flood susceptibility mapping using a novel ensemble weights-of-evidence and support vector machine models in GIS. J Hydrol

23. Strobl C, Malley J, Tutz G (2009) An introduction to recursive partitioning: rationale, application, and characteristics of classification and regression trees, bagging, and random forests. Psychol Methods 14(4):323–348

24. Gromping U (2009) Variable importance assessment in regression: linear regression versus random forest. Am Stat 63(4):308–319

25. James G, Witten D, Hastie T, Tibshirani R (2013) An introduction to statistical learning, pp 2197–4136. Springer

SMS Fraud Detection Using Machine Learning

Soumya Ranjan Prusty, Bhaskar Sainath, Suvendra Kumar Jayasingh, and Jibendu Kumar Mantri

Abstract The term SMS stands for short message service. It is a message networking method that uses smartphones and cell phones. It is a text messaging system that lets smart phones to communicate with one another. SMS has risen in prominence in current history, with firms adapting and optimizing SMS to be used as a service tool mostly because of its numerous benefits. We are using a unique technique that uses data mining and machine learning method to identify and filter spam communications in this study. We analyzed the characteristics of spam SMS in depth and discovered ten criteria that may effectively distinguish short message (or messaging) service spam from ham. If some adjustments are needed to identify a new scam, they were made manually, either by applying changes to current algorithms or by inventing new algorithms. In this method, as the quantity of clients and data grows, so does the amount of human work, which may also effectively separate spam text messages and ham SMS messages. The random forest classification method, we proposed, is the methodology that produced 99.9% accuracy.

Keywords SMS spam · Machine learning · Data preprocessing · Networking

1 Introduction

A message is transmitted digitally via the short messaging service (SMS), which is one of the most widely used communication services. Telecommunication firms have cut the price of SMS services, which has resulted in increasing SMS usage. These increases recruited hackers, resulting in an SMS spam problem. Any unsolicited message sent from a user's device is referred to as spam. Advertisements, free

S. R. Prusty
Institute of Management and Information Technology (IMIT) Cuttack, Cuttack, Odisha, India

B. Sainath
Jain University, Bangalore, Karnataka, India

S. K. Jayasingh · J. K. Mantri (✉)
Maharaja Sriram Chandra Bhanja Deo University, Baripada, Odisha, India
e-mail: jkmantri@gmail.com

services, rewards, and rewards are examples of spam communications. People prefer SMS messages to emails for communication because delivering SMS messages requires no Internet connection and is convenient and easy. The increased usage of text messaging and the issue of spams are becoming more prevalent. There seems to be a lot of security solutions available to mitigate the issue of SMS spam, although they are not yet developed. Many android applications exist on the play store to stop spam texts; however, due to lack of awareness, most people are unaware of them. Apart from applications, the existing monitoring solutions mostly concentrate on spam messages, as email is among the biggest problems, but even with the rise in the use of android platforms, SMS spam has become a big concern. Because mobile phones hold sensitive personal information such as card details, usernames, and passwords, SMS has been one of the cheapest ways to communicate and may be regarded as the easiest way to conduct phishing scams. Hackers are devising new ways of obtaining this data from smart phones, with SMS being one of the most straightforward. Type of phishing attack, or SMS-based hacking, is becoming increasingly prevalent these days. Phishing through SMS is becoming more common around the world, in which a person provides a malicious Web site by SMS and urges the recipient to click it, stealing important information from the recipient's smart phone. For identifying mobile fraud, there seems to be a variety of screening methods available, including smart cards, machine learning, fingerprints, matrix code scanner based, academic objectives, and identification based. SMS scammers may buy any contact information with just about any telephone number and use it to deliver spam messages, making it more difficult to track down the perpetrator. The top twenty spam detection area codes utilized by fraudsters were supplied by the US tango education center, in addition to the top ten SMS spam messages.

Cloudmark published a study in 2019 that detailed how hackers exploited SMS to deliver 876,000 spam messages as shown in Fig. 1. In 2020, the National Fraud Intelligence Bureau (NFIB) released a news report on most recent frauds, which has been examined by actions fraud. Phishers are attacking financial institution clients these days by delivering spam messages requesting for personal bank information, ATM pin number, and password. The major goal of our suggested strategy is to use machine

Fig. 1 Top five short message (or messaging) service spam messages

learning techniques to identify spam and ham SMS. For classification, we utilized a feature set of ten characteristics. These characteristics help distinguishing a spam SMS from a ham SMS. Machine learning techniques proved successful in avoiding zero-day threats and providing a high level of security in spam email filtering. The same method is being utilized for smart phones to eliminate short messaging spam; however, the characteristics of spam detection will differ from those of email spam since text messages are smaller and users use proper language for SMS.

2 Literature Review

There are a variety of text recognition text detection strategies accessible these days, such as blocking spam messages using mobile application and filter spam messages with classification techniques. In this paper, we will look at how to identify short messaging fraud applying machine learning approaches to filter spam messages depending on selecting features. El-Alfy and AlHasan [1] have suggested a short messaging filtering technique for both SMS and email. They looked at several approaches in order to come up with extracted features that would decrease difficulty. They utilized 13 characteristics, including URLs, probable spam terms, emotional icons, capital letters, wrinkly words, text data, JavaScript code, word structure, sender address, topic field, and junk mail domain, as well as two classification techniques, naive Bayes and support vector machine. Five email and SMS datasets were used to test their suggested model. For screening fraud communications, a messages subject structure (MTM) has been suggested by Jialin et al. [2]. To describe spam communications, the messaging predictive model (MTM) uses symbols, words, context terms, and subject terms and is predicated on a likelihood guess from latent semantic. They removed the scant difficulty by classifying SMS spam messages in to the unpredictable irregularity classes and afterward collecting all SMS phishing into a single document to detect keyword founder patterns using the k-means algorithm. The feature reweighting approach and the great word assault are two strategies for SMS spam detection that has been described by Chan et al. [3]. Both techniques concentrate on the text's size as well as the text's weight. The great word approach focuses on misleading the classifier's result by utilizing the fewest number of characters possible, whereas the characteristic reweighting technique has a novel parameterized function for resampling the dimensions. The experiment was assessed using two datasets: SMS and comments. Delany et al. [4] discussed the many techniques available for SMS spam filtering as well as the issues related with dataset gathering. They utilized ten groups to evaluate a huge dataset of SMS spam, including music, contests, dating, rewards, services, money, claim chatting, voice, and others. SMS spam messages were identified through information characteristics by Xu et al. [5]. For research experiment, they utilized two classification methods, SVM and k-nearest neighbors (KNN), as well as a feature set of factors, static, and temporal. They discovered that by integrating temporal and network characteristics SMS spam

messages may be identified more precisely and efficiently. Furthermore, they discovered methods to filter SMS spam messages by utilizing characteristics that comprise chart and time series, therefore eliminating the text's content. Text classification methods were tested on separate cell devices to see how well they filtered SMS spam by Nuruzzaman et al. [6]. All learning, screening, and updating operations were all carried out on a separate cell phone. Their suggested model was able to sort SMS spam with some excellent accuracy and low memory 20 N. Nuruzzaman et al. [6] have made significant utilization, and sufficient batch processing without requiring a huge dataset or any machine help.

Jayasingh et al. [7] have done weather prediction using hybrid soft computing models. Uysal et al. [8] presented a technique for SMS spam filtering that employs different feature selection methodologies, namely chi-square metrics and mutual information, to select information. They have created a real-time smartphone app for SMS spam detection based on the android platform. They utilized two Bayesian-based classification methods, Bayesian and binary. According to the authors, their suggested method detects both spam and genuine communications with great accuracy. For SMS spam detection, Yadav et al. [9] created a model text Assassin. They utilized a minimal feature set of 20 features as well as two machine learning algorithms, support vector machine (SVM) and Bayesian learning. They gathered a total of 2000 messages from visitors over the course of two months. In their suggested approach, anytime a user receives a message on mobile phone, SMS Assassin collects it without the user's awareness, retrieves feature values, and transmits them to the server. If the communications are marked as spam, the user will not be able to access them and they will be sent to the spam folder. The use of Bayesian filtering to identify SMS spam has been investigated by Hidalgo et al. [10]. They created two sets of data, one in English and the other in Spanish. According to their findings, Bayesian filters technologies that were previously used to identify spam emails may now be used to stop SMS spam. Cormack et al. [11] discussed feature engineering for mobile spam filtering. Jun et al. [12] explained the spam detection approach for secure mobile message communication using machine learning algorithms. Cormack et al. [13] explained the different types of machine learning algorithm. Jayasingh et al. [14] showed a novel approach for data classification using neural network.

3   Proposed Method

The suggested investigation in this work involves the procedure depicted in Fig. 2. The basic goal is to categorize text messaging either as ham or as spam.
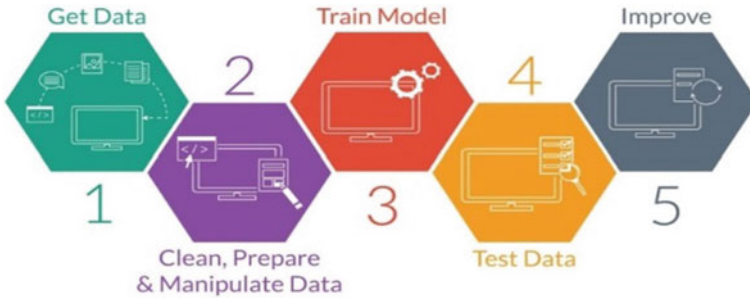
Fig. 2 Overview of approach for the model

Fig. 3 Sample dataset before cleaning

4 Dataset

We have taken the dataset having 5574 tagged data in the sample which contains 4827 ham messages and 747 spam messages. The dataset has two columns named ham and spam. It also has three unnamed columns as shown in Fig. 3.

Preprocessing, feature extraction, selection, and classification are all steps of spam detection as shown in Fig. 4.

5 Methodology

Our approach's major goal is to categories spam text messages as soon as they arrive on a smartphone, independently of whether they are newly produced spam messaging. We began by gathering data and finalizing the features for our trial. After finalizing the characteristics, we created a feature representation by extracting features from the messages (ham and spam). Several extracted features are used for training and testing. Our suggested system bases its judgment on ten characteristics.

The control structure of our suggested method is depicted. During the training process, the extracted features of phishing and ham texts are used to create a classifier

```
[n [8]:    1  df.head()
```

Out[8]:

	Type	Messages
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives aro...

Fig. 4 Sample dataset after cleaning

model. The classification evaluates if a new text is spam or not during the testing stage. Finally, classification of data for several machine learning techniques is done, and performance for each machine learning algorithm is assessed in order to choose the optimum methodology for our suggested strategy processing is the initial step inside the conversion of unstructured information to much more structured information, because characters are frequently substituted for terms in text messages. The stop word list extractor for English language, used in this research to eliminate stop words from SMS text messages depicts the frequency of terms in SMS messages, whereas the most familiar terms used in spam text messages belong to the noun and predicate categories. Similarly, pronouns, proposition, and a variety of many other stop words dominate the top terms in ham SMS messages.

6 Data Preprocessing

First, we converted the raw data into meaningful data. Then, we removed null values from our dataset, and then, we labeled the data into spam and ham. We lowered all the words and used NLTK stop words to remove the repeated word to reduce the noise. Then, we used porters Temer method to convert the word into vectors which have many dimensions because there are a lot of different words present in our dataset.

7 Algorithms

KNN algorithm

It is a supervised algorithm which can be used for regression and classification problems. Therefore, in businesses, it is mostly utilized to solve classification and

prediction issues. The two following characteristics will be a good way to describe KNN:

- KNN is a lazy learning algorithm since it does not have a learning and development period and instead utilizes all of the labeled training data and classification.
- KNN would also be a nonparametric learning method since it makes no assumptions regarding data.

Random forest Algorithms

It is a supervised algorithm which is used for classification problems. It mainly comprises of trees to give rise to a forest. The random forest constructs decision trees using datasets, finds the prediction in each, and votes for the best prediction. It is a based on integrated that is superior than just a decision tree classifier since it averages the results to minimize overfitting.

Naive Bayes Algorithms

The Bayes theorem is used in naive Bayes algorithms, which is a classification approach which focuses on the strong hypothesis that all predictions are autonomous of one another. A smartphone, for example, may be deemed intelligent if it has a touchscreen interface, Internet connectivity, and an excellent camera. Despite the fact that all of these characteristics are interdependent, they individually add to the likelihood that the smartphone.

Decision tree algorithm

An algorithmic technique that can partition the information in various ways based on different circumstances can be used to create decision trees. The much more powerful algorithm in the field of supervised learning algorithms is decision tree.

This can also be used for classification as well as regression. The decision nodes, in which the input is split, as well as the leaves, where we can get the result, are the two major components of a tree. A binary tree is used to forecast whether a person is healthy or unfit based on numerous factors such as age, diet and lifestyle, and daily exercise.

Confusion Matrix

The confusion matrix provides the details of prediction in a model as shown in Fig. 5. The **true positive** indicates that a positive value is predicted correctly [1]. A **false positive** occurs when the model forecasts the positive class erroneously as depicted. False positive indicates that a negative value is predicted as positive [0, 1]. **False negative** indicates that a positive value is predicted as negative [1, 0]. Always FN

Fig. 5 Confusion matrix

TN	FN
FP	TP

should be less in our model. **True negative** indicates that **v**alues that are actually negative and predicted also negative [0, 0].

Accuracy: It is the proportion of times the model correctly predicted the value. (AKA, the percentage of situations in which an industry's success or failure was correctly predicted). At first sight, this appears to be an excellent success statistic because the model was right the vast majority of the time. However, it is dependent on the situation.

The accuracy is defined by division of true positive + true negative by the sum of true positive + true negative + false positive + false negative.

Recall: It is the division of the true positive values by true positive and false negative values.

$$\text{Recall} = \text{TP}/\text{TP} + \text{FN}$$

Precision: It indicates the percentage of spam messages that the classification system correctly identified as spam. It demonstrates that everything is correct.

$$\text{Precision} = \text{TP}/\text{TP} + \text{FP}$$

8   Analysis

The execution of the proposed SMS spam detection system is evaluated through a series of tests. We first chose feature extraction methods on spam and ham message behavior and then separated this information from the data to create the classification model. To get system performance, different classification algorithms are used. They are support vector machine (SVM), random forest, AdaBoost, and decision tree.

It provides a built-in solution to cope with this issue. During split discovery, it overlooks missing data and then assigns them to the side that reduces the loss the greatest. If we do not want to utilize option and instead use alternative approaches, change the parameter to: Before starting the model prediction, we need to analyze the words, which words are repeated. The most common words are there in SMS, which are shown in Fig. 6. So, again we need to find common words in spam messages as shown in Fig. 7 and ham messages as shown in Fig. 8.

We have to find out the data which contains the information with a small number of potential values. Machine learning algorithms typically operate directly with categorical attributes, and in order to use them, subcategories must first be converted to integers before the deep learning model can be applied to them.

One-hot-encoder and updating values are two methods that may be used. To convert category variables into dummy/indicator variables, pandas is used to get dummy function. Imputation and erasure are two strategies for dealing with missing

```
In [233]: 1 wordcloud=WordCloud(background_color='white',width=1000,height=1000,max_words=50).generate(str(df['Messages']))
          2 plt.rcParams['figure.figsize']=(10,10)
          3 plt.title('Most common number of words in Messages')
          4 plt.axis('off')
          5 plt.imshow(wordcloud)
```

Fig. 6 Most common number of words in the message

```
In [234]: 1 wordcloud=WordCloud(background_color='white',width=1000,height=1000,max_words=50).generate(str(Spamfilter))
          2 plt.rcParams['figure.figsize']=(10,10)
          3 plt.title('Most common number of words in Messages')
          4 plt.axis('off')
          5 plt.imshow(wordcloud)
```

Fig. 7 Most common number of words in the spam message

```
In [235]: 1 wordcloud=WordCloud(background_color='white',width=1000,height=1000,max_words=50).generate(str(Hamfilter))
          2 plt.rcParams['figure.figsize']=(10,10)
          3 plt.title('Most common number of words in Messages')
          4 plt.axis('off')
          5 plt.imshow(wordcloud)
```

Fig. 8 Most common number of words in the ham message

values. Feature scaling is some of the most essential phases in the preprocessing of data before constructing a machine learning technique in machine learning algorithms. Many times, the dataset will contain features with a wide range of orders of magnitude, units, and ranges. Using unoptimized features, on the other hand, creates a difficulty in most machine learning applications. Machine learning methods that employ gradient descent as an optimization procedure, such as classification model and neural networks, require the dataset to be scaled. Whenever we employ proximity algorithms like random forest and SVM, which are most impacted by the set of advantages, we run into a similar dilemma. The reason for this is that

they use data point distances to evaluate how similar they are. We do not have to be concerned with its feature scalability. The size of the features has little effect on tree-based algorithms. The comparison of different algorithms is shown in Table 1.

It is observed from Table 1 and Fig. 9 that we have obtained the most accurate result by using random forests. Furthermore, using the random forest machine learning method, we were able to obtain 99.9% accuracy as shown in the graphical comparison of accuracy of different models in Fig. 9. During the training phase, the random forest classification method creates a collection of decision trees, which are subsequently used to classify randomly picked characteristics. The classification of ham and spam messages is done using the random forest (RF) algorithm. Random forest algorithm makes use of decision trees and eliminates the overfitting problem of the decision trees. Each tree present in the random forest algorithm provides its prediction, which vary from one another. The average of their performances is calculated. In the training

Table 1 Comparison of different algorithms on the basis of accuracy, precision, recall, and Roc-curve

Algorithms	Accuracy	Precision	Recall	Roc-curve
Random forest	99.99	99.98	99.99	99.89
Naïve Bayes	98.58	98.44	98.58	98.54
Decision tree	94.93	94.69	94.93	94.12
AdaBoost	92.98	92.53	92.98	92.54
SVM	90.59	90.27	90.59	90.11
KNN	96.01	96.00	96.01	95.89
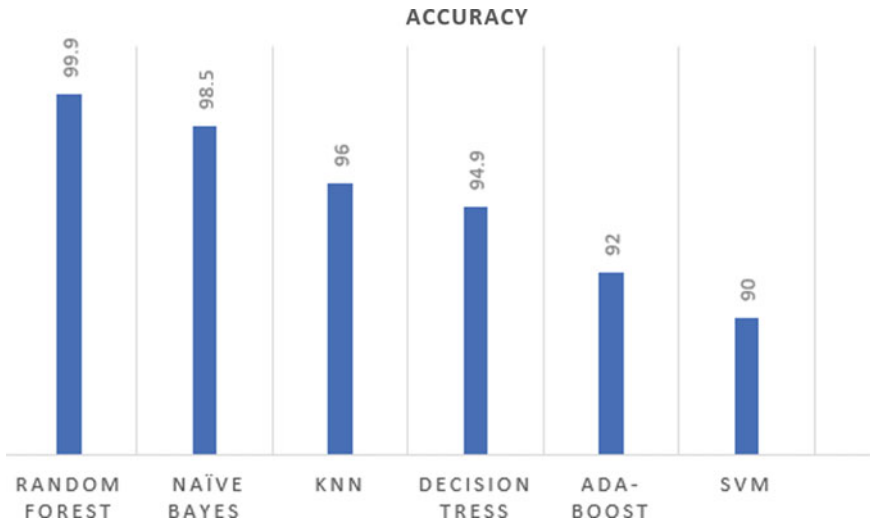
Fig. 9 Comparison of different algorithms on the basis of accuracy

phase, many decision trees are created and then they operate on a set of randomly selected features.

9 Conclusion and Future Work

As there is increase in usage of text messaging day by day, the problem of SMS spam is becoming more prevalent. Filtering SMS spam has been a major problem in recent years. In this research work, we have used six different machine learning algorithms such as decision tree, naive Bayes, random forest, KNN, SVM, and AdaBoost in which ten features from the dataset are used. There are 5574 tagged data in the sample, including 4827 messages belonging to ham messages and 747 messages belonging to spam messages collected. With accuracy of 99.9%, the random forest classification algorithm outperforms all other classification algorithms. The accuracy may further be enhanced by using hybrid soft computing models.

References

1. El-Alfy ESM, AlHasan AA (2016) Spam filtering framework for multimodal mobile communication based on dendritic cell algorithm. Future Gen Comput Syst 64:98–107. https://doi.org/10.1016/j.future.2016.02.018
2. Jialin M, Zhang Y, Liu J, Yu K, Wang X (2016) Intelligent SMS spam filtering using topic model. In: International conference on intelligent networking and collaborative systems (INCoS), pp 380–383. IEEE. https://doi.org/10.1109/INCoS.2016.47
3. Chan PPK, Yang C, Yeung DS, Ng WWY (2015) Spam filtering for short messages in adversarial environment. Neurocomputing 155:167–176. https://doi.org/10.1016/j.neucom.2014.12.034
4. Delany SJ, Buckley M, Greene D (2012) SMS spam filtering: methods and data. Expert Syst Appl 39:9899–9908. https://doi.org/10.1016/j.eswa.2012.02.053
5. Xu Q, Xiang EW, Yang Q, Du J, Zhong J (2012) SMS spam detection using non-content features. IEEE Intell Syst 27(6):44–51
6. Nuruzzaman MT, Lee C, Abdullah M, Choi D (2012) Simple SMS spam filtering on independent mobile phone. Secur Commun Netw 1209–1220. https://doi.org/10.1002/sec.577
7. Jayasingh SK, Mantri JK, Pradhan S (2021) Weather prediction using hybrid soft computing models. In: Udgata SK, Sethi S, Srirama SN (eds) Intelligent systems. Lecture notes in networks and systems, vol 185. Springer, Singapore. https://doi.org/10.1007/978-981-33-6081-5_4
8. Uysal AK, Gunal S, Ergin S, Gunal ES (2012) A novel framework for SMS spam filtering. In: International symposium on innovations in intelligent systems and applications (INISTA), pp 1–4. IEEE. https://doi.org/10.1109/INISTA.2012.6246947; Towards filtering of SMS spam messages using machine learning 29
9. Yadav K, Kumaraguru P, Goyal A, Gupta A, Naik V (2011) SMSAssassin: crowdsourcing driven mobile-based system for SMS spam filtering. In: 12th workshop on mobile computing systems and applications, pp 1–6. ACM. https://doi.org/10.1145/2184489.2184491
10. Hidalgo JMG, Bringas GC, Sánz EP, García FC (2006) Content based SMS spam filtering. In: ACM symposium on document engineering, pp 107–114. ACM. https://doi.org/10.1145/1166160.1166191

11. Cormack GV, Hidalgo JMG, Sánz EP (2007) Feature engineering for mobile (SMS) spam filtering. In: 30th annual international ACM SIGIR conference on research and development in information retrieval, pp 871–872. ACM. https://doi.org/10.1145/1277741.127795111
12. Jun LG, Nazir S, Khan HU, Ul Haq A (2020) Spam detection approach for secure mobile message communication using machine learning algorithms. Secur Commun Netw 2020:6. Article ID 8873639. https://doi.org/10.1155/2020/8873639
13. Cormack GV, Hidalgo JMG, Sánz EP (2007) Spam filtering for short messages. In: 16th ACM conference on conference on information and knowledge management, pp 313–320. ACM. https://doi.org/10.1145/1321440.1321486
14. Jayasingh SK, Gountia D, Samal N, Chinara PK (2021) A novel approach for data classification using neural network. IETE J Res. https://doi.org/10.1080/03772063.2021.1986152

Differential Evolution-Improved Dragonfly Algorithm-Based Optimal Radius Determination Technique for Achieving Enhanced Lifetime in IoT

J. Sengathir, M. Deva Priya, A. Christy Jeba Malar, and G. Sandhya

Abstract Internet of Things (IoT) has practically facilitated the option of connecting devices that necessitates cooperation through the Internet. The IoT environment comprises of self-configuring and smart objects that have the possibility of interacting with one another using the infrastructure of a global network. Clustering is the predominant approach that potentially works on the improvement of network lifetime. In this paper, a new clustering scheme using Differential Evolution-Improved Dragonfly Algorithm-based Optimal Radius Determination Strategy (DE-IDAORDS) is proposed for achieving prolonged lifetime in IoT. DE-IDAORDS facilitates the selection of Cluster Head (CH) based on the hybridized merits of DE and IDA for balancing the trade-off between exploration and exploitation to achieve effective clustering that attributes toward excellent energy stability and network lifetime. It adopts fitness function evaluation using parameters like cluster radius, distance and energy during CH selection. The results of the proposed DE-IDAORDS confirm better results in terms of energy, cost function, and number of alive nodes in contrast to the benchmarked schemes taken for investigation. The results also demonstrate that the proposed DE-IDAORDS scheme is capable of adequately enhancing the convergence rate and energy conservation in a minuscule period.

J. Sengathir
Department of Information Technology, CVR College of Engineering, Mangalpally, Vastunagar, Hyderabad, Telangana, India

M. Deva Priya (✉)
Department of Computer Science and Engineering, Sri Eshwar College of Engineering, Coimbatore, Tamilnadu, India
e-mail: devapriya.m@sece.ac.in

A. Christy Jeba Malar
Department of Information Technology, Sri Krishna College of Technology, Kovaipudur, Coimbatore, Tamilnadu, India
e-mail: a.christyjebamalar@skct.edu.in

G. Sandhya
Department of Computer Science and Engineering, Sri Krishna College of Technology, Kovaipudur, Coimbatore, Tamilnadu, India
e-mail: sandhya.g@skct.edu.in

1 Introduction

In the recent past, Internet of Things (IoT) has developed tremendously owing to its
wide use across the world. It is a technological gift that is developed to overcome the
challenges of non-ubiquitous sensing and non-pervasive computing enabled using
Wireless Sensor Network (WSN) technologies [1–3]. It includes a collection of
assorted devices that are openly connected to the Internet. IoT links the devices
directly to the network through universal infrastructure, where the communication
amid devices is through smart and self-configuring devices. The smart association
of devices involving network resources plays a dominant role in IoT. The data accu-
mulated from diverse devices are analyzed to disseminate the valuable data through
applications built to deal with the demands [4]. The devices are capable of dealing
with the breach amid physical and digital worlds, facilitating fresh kind of services.
Though IoT is used in diverse applications, it involves many challenges which
include incorporation of numerous technologies and standards that focus on sensing,
computational competences, storage, and connectivity [5, 6]. The nodes are resource
constrained with restricted energy, memory, computation, and communication abil-
ities. The nodes may also act as gateways to distant users. Owing to insufficient
supply of energy to nodes, energy preservation is predominant in attaining extended
network lifetime. Clustering is effective as it improves scalability of network by
dropping channel contention and packet collision, and enhancing throughput. Profi-
cient Cluster Head (CH) selection increases the load efficiency on CHs as energy
may be conserved and network lifetime may be improved [7].

In this paper, a new clustering scheme using Differential Evolution (DE)-
Improved Dragonfly Algorithm-based Optimal Radius Determination Strategy (DE-
IDAORDS) is proposed for achieving prolonged lifetime in IoT. It integrates the
exploration capabilities of DE with the exploitation potentialities of improved DFA
to establish balanced deviation between the local and global search process. It
includes the parameters of cluster radius, distance and energy during CH selec-
tion to evaluate the fitness function under optimization. The simulation experiments
of the proposed DE-IDAORDS are conducted using the evaluation metrics such as
throughput, normalized energy, time incurred for CH selection, cost function and
number of alive nodes for varying number of rounds and iterations in contrast to the
benchmarked schemes taken for investigation.

2 Related Work

In this section, the work done by various authors related to CH selection are discussed. Feng et al. [8] have propounded an enhanced K-means algorithm to cluster nodes and perform weighted assessment to enhance cluster construction. The cluster is divided or broken based on the assessment outcomes to get non-uniform clustering. The authors have focused on increased transmission delay due to data fusion and have built a delay-enhanced fusion tree. Parent nodes are chosen based on distance and energy. Time slot distribution is enhanced by building a fusion tree, and the delay involved in transmission is reduced. Halder et al. [9] have dealt with improving the network lifespan by balancing energy depletion amid CHs. The radius at every level plays a dominant role in improving network lifetime. Lifetime Maximizing optimal Clustering Algorithm (LiMCA) is propounded for energy stringent IoT devices. Mainly, LiMCA involves a stochastic positioning mechanism for nodes in the network. A protocol is designed to train nodes about coarse-grain positions. Reddy and Babu [10] have adopted a hybrid model using Moth Flame Optimization (MFO) and Ant Lion Optimization (ALO) to increase the performance of CH selection amid devices. The distance and delay are maintained, and the temperature and load are balanced for achieving ideal CH selection. The mechanism focuses on convergence, maintaining alive nodes, controlling of energy, load and temperature. Reddy and Babu [11] have designed a Self-Adaptive Whale Optimization Algorithm (SAWOA) for ensuring energy-based selection of CH and clustering in Wireless Sensor Network (WSN)-based IoT. The propounded schemes offer better network lifetime. Dhumane and Prasad [12] have propounded Multi-Objective Fractional Gravitational Search Algorithm (MOFGSA) for finding optimal CHs that support energy effective routing in an IoT network. The CHs are assessed based on a fitness function in terms of distance, lifespan, delay, and energy.

Srinidhi et al. [13] have propounded Hybrid Energy Efficient and QoS Aware (HEEQA) scheme, an amalgam of Quantum Particle Swarm Optimization (QPSO), and enhanced Non-dominated Sorting Genetic Algorithm (NSGA) to attain energy balance amid devices. The parameters in MAC layer are altered to decrease energy consumption of devices. NSGA focuses on multi-objective optimization, while the QPSO algorithm supports in finding the optimum cooperative grouping. Govindaraj and Deepa [14] have dealt with the optimization of IoT in WSNs that focuses on handling energy and accuracy involved in complex clustering mechanisms. Capsule Neural Network (CNN)-based learning model that manages network energy at an optimal level is propounded. This architecture supports efficient routing and optimization, wherein activation is done during the forward pass. Sennan et al. [15] have propounded Type-2 Fuzzy Logic-based Particle Swarm Optimization (T2FL-PSO) algorithm to choose the optimum CH to prolong the lifespan of a network. This scheme is useful in offering precise solution in indeterminate network surroundings. Residual energy along with the distances amid the nodes and the Base Station (BS) are used to find the fitness. Virtual clusters are built depending on the distance amid nodes and the CH or the BS. Dev et al. [16] have propounded a scheme wherein the

CH is chosen by the Over taker Assisted Wolf Update (OA-WU) that is a combination of the conceptions of Gray Wolf Optimization (GWO) algorithm and Rider Optimization Algorithm (ROA). This scheme is based on distance, radius and energy of the cluster. Energy is conserved, and convergence rate is improved in a short duration.

3 Proposed DE-IDAORDS Scheme

The network model considered during the implementation of the proposed DE-IDAORDS scheme consists of a single BS and finite number of static sensor nodes. The communication in the network always occurs within the range facilitated by the given radio. The network is assumed to face challenges related to energy consumption, data sensing, sensor allocation, radio communication and topology characteristics. The sensor nodes in the network are localized in the application areas either manually or randomly. However, extending network lifetime is a herculean task in WSNs enabled IoT. In this context, clustering is indispensable for partitioning the sensor nodes to construct clusters. Moreover, the transmission of data from one node to another is a challenging task in WSN-enabled IoT. This challenge of data transmission needs to be overcome based on the determination of optimal shortest paths. In this context, several works focusing on the establishment of different system models that aid in attaining the distribution of data packets among the cluster members and BS using the merits of routing protocols are contributed to the literature. However, energy and location of sensor nodes are identified as prime factors for performing CH selection in hierarchical routing. In this paper, the proposed DE-IDAORDS scheme is presented as a potential CH selection strategy using the optimizing factors of cluster radius, energy, and distance for achieving energy stability and prolonged network lifetime.

3.1 Cluster Radius Estimation

The factor of cluster radius needs to be estimated for enhancing the network lifespan in the clustering architecture. This cluster radius (C_R) is estimated by dividing the complete network area of radius (R_N) into 'k' ring sectors based on the angle of disk sector (φ). In the network model, the sensor nodes are independent of the size of the network and transmit data to the neighboring CHs within a fixed radius. In specific, each node possesses a particular sensing and communication radius to estimate its network coverage. But communication radius depending on the number of neighboring nodes needs to be essentially computed for reliable data transmission. To achieve the estimation, potential optimization strategy becomes essential, and thus, DE-IDA mechanism is employed for selecting the optimal cluster radius depending on the fitness function 1 as specified in Eq. (1).

$$F_1 : C_R = \min_i \frac{\text{SN}_{R-\text{UC}(i)}}{\text{SN}_{\text{Total}}} \tag{1}$$

where 'SN$_{R-\text{UC}(i)}$' and 'SN$_{\text{Total}}$' represent the number of sensor nodes that are not covered by the radius and the total number of nodes existing in the network.

3.2 Energy Consumption

Adequate amount of energy is essential for data transmission. This energy is essential for carrying out the functions of sensing, aggregation, transmission and reception. The cumulative energy $\left(E_{\text{SN}}^{\text{Initial}}\right)$ necessary for communication is the aggregate sum of the energy spent for idle state $\left(E_{\text{SN}}^{\text{IS}}\right)$, data reception $\left(E_{\text{SN}}^{\text{DR}}\right)$, electronic energy $\left(E_{\text{SN}}^{\text{EE}}\right)$, and data transmission $\left(E_{\text{SN}}^{\text{DT}}\right)$ as specified in Eq. (2).

$$F_2 : E_{\text{SN}}^{\text{Initial}} = E_{\text{SN}}^{\text{IS}} + E_{\text{SN}}^{\text{DR}} + E_{\text{SN}}^{\text{EE}} + E_{\text{SN}}^{\text{DT}} \tag{2}$$

The above-mentioned energy consumption is the second factor (fitness function 2) to be optimized during the CH selection process.

3.3 Distance Estimation

In the process of clustering, CH is responsible for generating and forwarding an advertisement message to the complete set of nodes existing in the network to inform their role during the communication process. Whenever a sensor node receives an advertisement message from a CH, it commences to estimate the distance between itself and the CH. Moreover, the construction of clusters within the network is facilitated by the sensor nodes that possess the least distance with the CH. This distance measure is determined and represented as a matrix as shown in Eqs. (3) and (4).

$$d_{M(a,b)} = \sqrt{(A_W - B_W)^2 + (A_N - B_N)^2} \tag{3}$$

$$F_3 : D_{M(i \times j)} = \begin{bmatrix} d_{M(\text{CH}_1 - S_{11})} & d_{M(\text{CH}_1 - S_{21})} & \cdots & d_{M(\text{CH}_1 - S_{M1})} \\ d_{M(\text{CH}_2 - S_{11})} & d_{M(\text{CH}_2 - S_{21})} & \cdots & d_{M(\text{CH}_2 - S_{M1})} \\ \cdots & \cdots & \cdots \cdots \\ d_{M(\text{CH}_K - S_{11})} & d_{M(\text{CH}_K - S_{21})} & \cdots & d_{M(\text{CH}_K - S_{M1})} \end{bmatrix} \tag{4}$$

The complete objective of the proposed DE-IDAORDS-based IoT clustering model is defined in Eq. (5).

$$\text{Obj}_{\text{FN}} = \text{Min}\left(F_1 + \frac{1}{F_2} + F_3 \right) \tag{5}$$

where 'F_1', 'F_2', and 'F_3' represent the fitness function parameters considered for optimization with cluster radius set in the range from 0 to 50.

3.4 Dragonfly Algorithm (DFA)

Primitive DFA algorithm is proposed based on the inspiration derived from the dragonflies' social behavior with respect to the hunting and migration for food [17]. In specific, the dragonflies' hunting phenomenon mimics the random movement in the local search space as it corresponds to the exploitation potentialities of the algorithm. On the other hand, migration of dragonflies is like the exploration capability that investigates the solution in the entire problem domain. In this context, the feasible solution in the problem domain with respect to optimal radius-based CH selection (population matrix) is shown in Eq. (6).

$$S_i^D = \{S_1, S_2, \ldots, S_N\} \tag{6}$$

where 'S_i^D' represents the position of the search agent (dragonfly) with 'D' and '$\hat{N}$' as the dimensional search space and search agents' count respectively. The searching process of the DFA algorithm comprises of the parameter of separation, cohesion and alignment determined for '$\hat{N}$' individual neighbors. In specific, the parameter separation and cohesion with respect to DFA are computed based on Eqs. (7) and (8).

$$S_{P(i)} = -\sum_{j=1}^{\hat{N}} \left(S - \hat{S}_j \right) \tag{7}$$

$$C_{P(i)} = \frac{\sum_{j=1}^{\hat{N}} \left(S - \hat{S}_j \right)}{N} - S \tag{8}$$

On the other hand, the factor of alignment is determined based on Eq. (9).

$$A_{P(i)} = \frac{\sum_{j=1}^{\hat{N}} V_j}{N} \tag{9}$$

where 'i' and 'j' represent the current and neighboring individual solutions considered from the search space for exploration and exploitation. Moreover, 'V_j' highlights the velocity associated with the jth neighboring individual solution.

Further, the degree of attraction $(A_{R(i)})$ and distraction $(D_{R(i)})$ of the search agents toward the determination of optimal solution are computed as shown in Eqs. (10) and (11).

$$A_{R(i)} = S^+ - S \tag{10}$$

$$D_{R(i)} = S^- - S \tag{11}$$

where 'S' represents the current position of search agent from which the required solution is closer (S^+) and farther (S^-) depending on the position toward solution determination.

At this juncture, the step vector '$\Delta S_{P(t+1)}$' is updated based on Eq. (12).

$$\Delta S_{P(t+1)} = (\alpha S_{P(i)} + \beta C_{P(i)} + \gamma A_{P(i)} + \delta A_{R(i)} + \mu D_{R(i)}) \\ + W_{\text{Intertial}} * \Delta S_{P(t)} \tag{12}$$

where 'α', 'β', 'γ', 'δ', and 'μ' represent the factors of separation, cohesion, alignment, attraction and distraction for achieving better exploration and exploitation with 'W_{Inertial}' as the inertial weight. Moreover, the position of the search agent is revised based on Eq. (13).

$$S_{P(t+1)} = S_{P(t)} + \Delta S_{P(t+1)} \tag{13}$$

In addition, the search agents' position (dragonfly) using the merits of Levy fight function is updated using Eq. (14).

$$S_{P(t+1)} = S_{P(t)} + \text{Levy}\,(d)\hat{S}_{P(t)} \tag{14}$$

Finally, the fitness function presented in Eq. (5) is evaluated based on the determined position vector until the termination condition is satisfied.

3.5 Differential Evolution

Differential evolution is proposed as a potential optimization algorithm that facilitates significant results over different linear objective functions which are like objective functions formulated for CH selection [18]. It offers better convergence rate and aids in attaining better global optimal solution during searching. In this proposed DE-IDAORDS scheme, DE is specifically utilized for handling the computation intensive cost functions with minimized number of convergence properties and control variables. It also adopts random mutation that inherits the weighted difference estimated between two vectors. It inherits trial factor for employing mutation and crossover

over the target vector using mutant vector specified in Eq. (15).

$$m_v^i = S_{r_1} + M_F\left(S_{r_2} - S_{r_3}\right) \tag{15}$$

where 'r_1', 'r_2', and 'r_3' represent the randomly selected values that highlight the individuals in 'N' dimensions. Moreover, 'M_F' represents the component of crossover introduced for refining the algorithmic diversity.

Further, the vectors of 'm_v^i' and 'S_P^i' are built using crossover for constructing the trial vector based on Eq. (16).

$$U_{\text{Cross}}^i = \begin{cases} m_v^i & \text{if } (\text{rnd}_i \le C_F) \text{ or } \left(R_{\text{CF}} = R_{\text{CF}}^{\text{Rnd}}\right) \\ S_v^i & \text{if } (\text{rnd}_i > C_F) \text{ or } \left(R_{\text{CF}} \neq R_{\text{CF}}^{\text{Rnd}}\right) \end{cases} \tag{16}$$

where 'C_F' and 'R_{CF}' represent the crossover factor and dimension considered for searching in the range of 0 and 1. Moreover, $R_{\text{CF}}^{\text{Rnd}} \in [1, D]$ highlights the randomly selected individual index value. In addition, the fitness value of the objective function (specified in Eq. (5)) is utilized for selecting the optimal solution from the target and trial vector.

3.6 Hybridization of DE and DFA

The primitive aim of hybrid optimization is to establish superior balance between exploration and exploitation that attributes toward the mitigation of limitations possessed in the parent optimization method for attaining enhanced optimal solution. The traditional DFA algorithm possesses exploration potential based on the randomness of the initially generated population depending on the significance of Levy fight search to achieve solution diversity. However, DFA does not inherit any mechanism for storing the best solution. It also discards the local best values during implementation when it exceeds the global best solution. It is also considered to be slow and suffers from the problem of premature convergence. This limitation completely fails in tracking the comprehensive set of solutions that introduces the possibility of converging the solution to a global optimum. Thus, the hybridization of DE and DFA concentrates on the performance improvement by (i) including individual particle memory that stores the global best solution along with the local best solution for attaining the global optimal point, (ii) integrating iteratively with DE to execute the set of solutions determined by DFA in a predominant manner, and (iii) hybridizing improved learning-based mutation of DE into DFA for achieving population diversity. It also introduces the evaluation of convergence power using local and global best solutions that control the span of convergence. In addition, it utilizes the convergence power to generate a new set of population when it exceeds the specific threshold. In this process of hybridization, mutation inherent in DE is

carried out using the local and global best solutions for identifying the predominant solutions in the search space. This mutation process adopts an enhanced learning strategy to include the local and best solutions that improve the population diversity. Then, the vector of target associated with DE is improved with the iterative best solution identified from DFA depending on Eq. (17).

$$m_v^{(i,k)} = S^{(i,k)} + \text{rnd}_i \left(S^{(i,k)} - S_{\text{Old}}^{(i,k)} \right) + M_F^1 \left(s_{\text{Leader}} - S^{(i,k)} \right)$$
$$- M_F^2 \left(r_n * s_{\text{Worst}} - S_{\text{LBest}}^{(i,k)} \right) \tag{17}$$

where '$S_{\text{Old}}^{(i,k)}$', '$S_{\text{LBest}}^{(i,k)}$', and 's_{Leader}' represent the positions of the individual solution and global worst particle in the preceding iteration. In this case, 'rnd_i' represents the random number that lies between 0 and 1 which obeys the characteristics of normal distribution '$N(0, 1)$'. Moreover, the values of 'M_F^1' and 'M_F^2' are determined based on Eqs. (18) and (19).

$$M_F^1 = M_F + \text{rnd}_1 M_F \tag{18}$$

$$M_F^2 = M_F - \text{rnd}_1 M_F \tag{19}$$

At this juncture, the value of 'M_F' representing the mutation factor is computed based on Eq. (20).

$$M_F = 0.8 + \frac{(0.8 - 0.2)\text{Iter}_{\text{Curr}}}{\text{Iter}_{\text{Max}}} \tag{20}$$

where '$\text{Iter}_{\text{Curr}}$' and '$\text{Iter}_{\text{Max}}$' represent the current iteration and maximum number of iterations considered during the implementation of DE-IDAORDS.

Finally, the proposed DE-IDAORDS-based IoT clustering optimization model is adopted for clustering to improve the network lifetime and energy stability.

4 Results and Discussion

The performance evaluation of the proposed DE-IDAORDS-based IoT clustering model and the benchmarked approaches is conducted using MATLAB R2018a. The data acquisition associated with the implemented IoT environment is considered from the data science community of Kaggle. This experimental investigation is carried out based on the time taken for CH selection, cost function, normalized energy, alive nodes and throughput.

The network area considered for implementation is 100×100 square meters in which 100 sensor nodes are deployed randomly throughout the entire region.
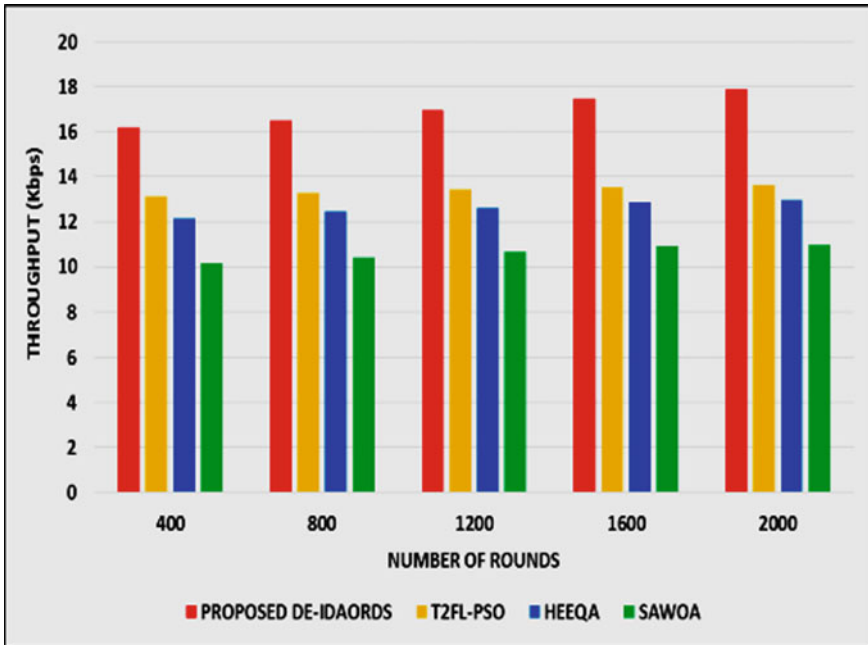
Fig. 1 Throughput for Varying Number of Rounds

The initial energy of sensor nodes is set to 0.5 J with simulation rounds of 2000 and packet size of 4500 bits. Initially, Figs. 1 and 2 demonstrate the throughput and normalized energy realized in the network during the implementation of the proposed DE-IDAORDS and the benchmarked T2FL-PSO, HEEQA, and SAWOA approaches for varying number of rounds. The results evidently prove that the proposed DE-IDAORDS scheme is potent in estimating the optimal cluster radius to a more accurate level and confirm better throughput by propagating the packets from the source to the destination nodes. On the other hand, the normalized energy sustained in the network is maximized by the proposed DE-IDAORDS by adopting the potentialities of trial and target vector during local and global searches. Hence, the normalized energy used by the proposed DE-IDAORDS for varying number of rounds is comparatively improved by 13.29%, 15.64% and 17.46% when compared to the benchmarked T2FL-PSO, HEEQA and SAWOA approaches. Moreover, the proposed DE-IDAORDS for varying number of rounds reduces the normalized energy by 12.98%, 14.82% and 16.71% in contrast to the benchmarked T2FL-PSO, HEEQA and SAWOA approaches.

Further, Figs. 3 and 4 depict the cost function and the time incurred for CH selection by the proposed DE-IDAORDS and the benchmarked T2FL-PSO, HEEQA, and SAWOA approaches for varying number of iterations. The results confirm that the proposed DE-IDAORDS scheme is capable of minimizing the cost function as it integrates the merits of DE and DFA toward optimal CH selection process.
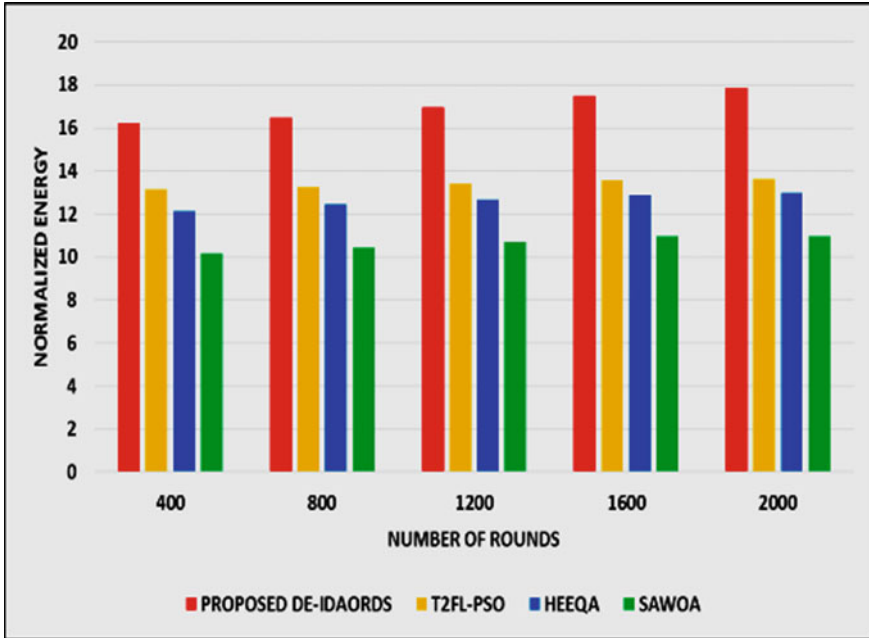
Fig. 2 Normalized Energy for Varying Number of Rounds
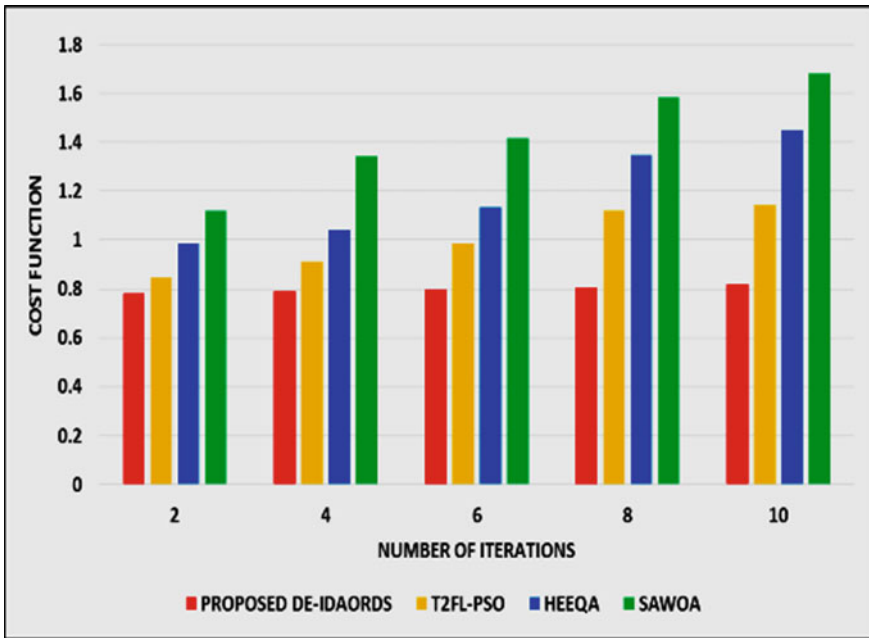
Fig. 3 Cost Function for Varying Number of Rounds
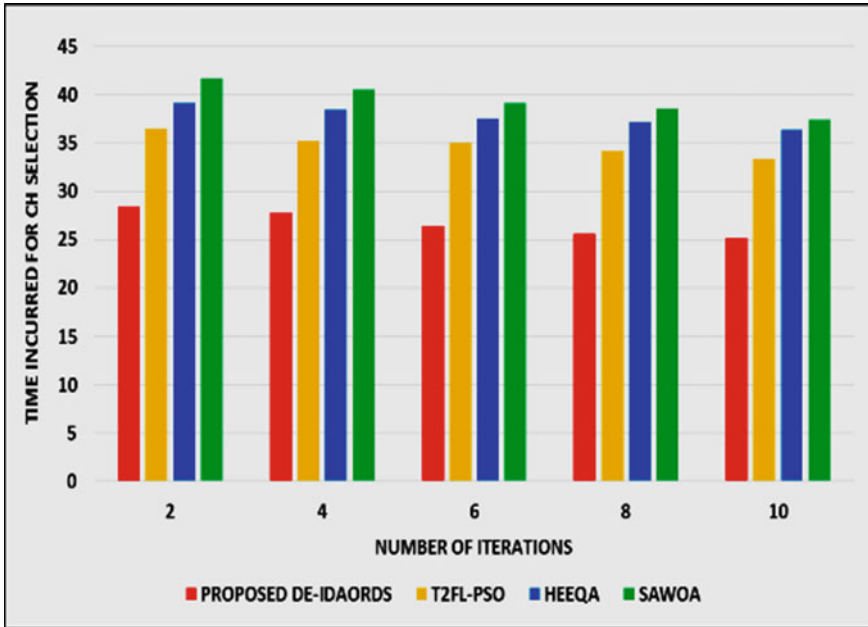
Fig. 4 Time incurred for CH Selection for Varying Number of Rounds

The time incurred for CH selection as attained by the proposed DE-IDAORDS is comparatively minimized since it adopts the adjustment of migration factor of DFA into the mutation factor of DE toward better exploration and exploitation. Thus, the cost function attained by the proposed DE-IDAORDS for varying number of iterations is comparatively reduced by 17.21%, 19.86% and 21.94% when compared to the benchmarked T2FL-PSO, HEEQA and SAWOA approaches.

Moreover, the proposed DE-IDAORDS for varying number of iterations minimizes the time incurred for CH selection by 14.29%, 16.84% and 19.23% when compared to the benchmarked T2FL-PSO, HEEQA and SAWOA approaches. In addition, Fig. 5 depicts the number of alive nodes in the network as attained by the proposed DE-IDAORDS and the benchmarked T2FL-PSO, HEEQA and SAWOA approaches for varying number of rounds. The number of alive nodes sustained in the network is improved due to the inclusion of DE that adaptively helps the DFA in exploring the search space during the optimal CH selection process. The proposed DE-IDAORDS for varying number of rounds maximizes the number of alive nodes by 12.98%, 14.76% and 16.21% when compared to the benchmarked T2FL-PSO, HEEQA and SAWOA approaches.
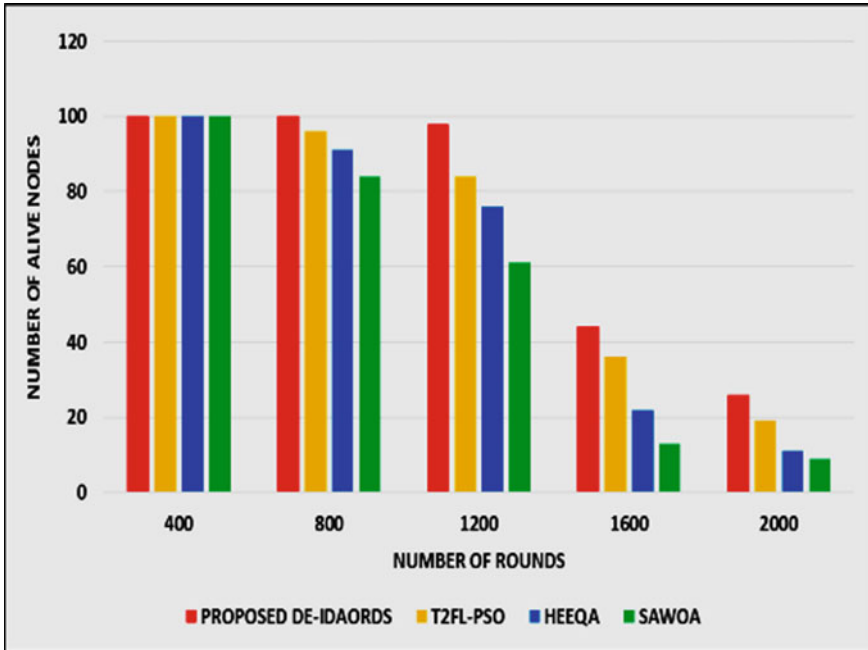
Fig. 5 Number of Alive Nodes for varying Number of Rounds

5 Conclusion

In this paper, the proposed DE-IDAORDS scheme is a significant scheme for selecting optimal CHs through the exploration and exploitation benefits of DE and DFA algorithms. It chooses optimal CH using the optimization factors of cluster radius, energy model and distance measures. It facilitates optimal tuning of the process of determining the cluster radius. The simulation results confirm that the cost function attained by the proposed DE-IDAORDS for varying number of iterations is comparatively reduced by 17.21%, 19.86% and 21.94% when compared to the benchmarked T2FL-PSO, HEEQA and SAWOA approaches. Further, the proposed DE-IDAORDS for varying number of iterations minimizes the time incurred for CH selection by 14.29%, 16.84% and 19.23% when compared to the benchmarked T2FL-PSO, HEEQA and SAWOA approaches. In addition, the results evidently prove that the proposed DE-IDAORDS for varying number of rounds maximizes the number of alive nodes by 12.98%, 14.76% and 16.21% in contrast to the benchmarked T2FL-PSO, HEEQA and SAWOA approaches. As a part of the future plan, it is decided to formulate a Mayfly-based optimal CH selection to improve the network lifetime in IoT on par with the proposed DE-IDAORDS scheme.

References

1. Behera TM, Mohapatra SK, Samal UC, Khan MS, Daneshmand M, Gandomi AH (2019) Residual energy-based cluster-head selection in WSNs for IoT application. IEEE Internet Things J 6(3):5132–5139
2. Balamurugan A, Priya M, Janakiraman S, Malar A (2021) Hybrid stochastic ranking and opposite differential evolution-based enhanced firefly optimization algorithm for extending network lifetime through efficient clustering in WSNs. J Netw Sys Manag 29(3):1–31
3. Balamurugan A, Janakiraman S, Priya DM (2022) Modified African buffalo and group teaching optimization algorithm-based clustering scheme for sustaining energy stability and network lifetime in wireless sensor networks. Trans Emerg Telecommun Technol 33(1)
4. Alazab M, Lakshmanna K, Reddy T, Pham QV, Maddikunta PKR (2021) Multi-objective cluster head selection using fitness averaged rider optimization algorithm for IoT networks in smart cities. Sustain Energy Technol Assess 43:100973
5. Janakiraman S, Priya DM (2020) An energy-proficient clustering-inspired routing protocol using improved Bkd-tree for enhanced node stability and network lifetime in wireless sensor networks. Int J Commun Sys 33(16):e4575
6. Janakiraman S, Priya M, Devi SS, Sandhya G, Nivedhitha G, Padmavathi S (2021) A Markov process-based opportunistic trust factor estimation mechanism for efficient cluster head selection and extending the lifetime of wireless sensor networks. EAI Endorsed Trans Energy Web e9
7. Rajarajeswari PL, Karthikeyan NK, Priya MD (2015) EC-STCRA: Energy Conserved-Supervised Termite Colony based Role Assignment scheme for Wireless Sensor Networks. Procedia Comp Sci 57:830–841
8. Feng X, Zhang J, Ren C, Guan T (2018) An unequal clustering algorithm concerned with time-delay for internet of things. IEEE access 6:33895–33909
9. Halder S, Ghosal A, Conti M (2019) LiMCA: an optimal clustering algorithm for lifetime maximization of internet of things. Wireless Netw 25(8):4459–4477
10. Reddy MPK, Babu MR (2019) A hybrid cluster head selection model for internet of things. Clust Comput 22(6):13095–13107
11. Reddy MPK, Babu MR (2019) Implementing self-adaptiveness in whale optimization for cluster head section in internet of things. Clust Comput 22(1):1361–1372
12. Dhumane AV, Prasad RS (2019) Multi-objective fractional gravitational search algorithm for energy efficient routing in IoT. Wireless Netw 25(1):399–413
13. Srinidhi NN, Lakshmi J, Kumar SD (2019) Hybrid energy efficient and QoS aware algorithm to prolong IoT network lifetime. In: International conference on ubiquitous communications and network computing, pp 80–95. Springer, Cham
14. Govindaraj S, Deepa SN (2020) network energy optimization of IOTs in wireless sensor networks using capsule neural network learning model. Wireless Pers Commun 115(3):2415–2436
15. Sennan S, Ramasubbareddy S, Balasubramaniyam S, Nayyar A, Abouhawwash M, Hikal NA (2021) T2FL-PSO: type-2 fuzzy logic-based particle swarm optimization algorithm used to maximize the lifetime of internet of things. IEEE Access 9:63966–63979
16. Dev K, Poluru RK, Kumar RL, Maddikunta PKR, Khowaja SA (2021) Optimal radius for enhanced lifetime in IoT using hybridization of rider and grey wolf optimization. IEEE Trans Green Commun Netw 5(2):635–644
17. Mirjalili S (2016) Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. Neural Comput Appl 27(4):1053–1073
18. Storn R, Price K (1997) Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 11(4):341–359

Optimization of Operating Parameters for Improve the Combustion in Single Cylinder Four Stroke DICI VCR Engine Using Grey Relation Analysis

Krushnashree Sushree Sangita Sahoo, Dr. Anand Gupta, and Amritam Mohapatra

Abstract   This paper aims to present a novel method for the optimization of operating parameters of a four stroke variable compression ratio direct injection compression ignition engine with multi objective-based orthogonal design with grey relation analysis. Testing is conducted on the test engine. Combustion testes are carried out on the test engine using pure commercial diesel fuel under atmospheric conditions. In the present work, operating parameters such as compression ratio, injection timing and injection pressure are optimized considering the multi responses combustion parameters such as peak pressure, rate of pressure rise, net heat release rate and mean gas temperature. The Grey relation grade (GRG) is determined for optimal set of operating parameters. For the validation of the result, a confirmation test is carried out. The experiment result shows that the set of operating parameters improve the combustion in direct injection compression ignition (DICI) engine.

1   Introduction

There are various application in which diesel engines are used such as transportation, construction activities, as these engine is having high power to weight ratio, in addition to that its also having low specific fuel consumption [1, 2]. The price and demand of fossil fuel is increasing day by day and the controls on these are very difficult. This brings interest to the researchers working on these areas to find an innovative way to control the price and demand. As the millions of existing compression ignition engine presently running, so the major modification on the existing engine is very difficult. In the present study's author is identifying the set of operating parameters on which the combustion on the engine is improved without any modification on the existing engine. For improving the combustion characteristics biodiesel with improve fuel

K. S. S. Sahoo · Dr. A. Gupta (✉) · A. Mohapatra
Department of Mechanical Engineering, Indira Gandhi Institute of Techbology, Sarang, India
e-mail: anandgupta@igitsarang.ac.in

Table 1 Operating parameters and their level for commercial diesel

Factor	Description	Level 1	Level 2	Level 3
CR	Compression ratio	12	14	18
IT	Injection timing (degree before TDC)	10	15	20
IP	Injection pressure (MPa)	30	35	40

properties used [3]. FEM analysis of critical engine component can also improve the performance of internal combustion engine [4]. Various optimization techniques are used to optimize the operating parameter for good combustion and performance characteristics [5–10].

2 Material and Method

Prior to do the optimization a number of experiments is carried out using commercial diesel in the experimental setup. For Taguchi optimization first a set of operating parameters and their levels are chosen are given in Table 1. Taguchi design of experiment (DOE) is formed using Minitab 16 software. The experiments [11, 12] are conducted based on Taguchi L_9 design of experiment. As Taguchi method is used to optimize the single objective function. An algorithm Grey relation analysis is used to solve multi objective problem.

2.1 Experimental Work

The experimental setup is shown in Fig. 1. The DOE is formed using three operating parameters and three levels. The experimental [13] are conducted using the setup based on Taguchi L_9 orthogonal array. Taguchi justify the uses of orthogonal design to reduce the number of experiment. The experimental data based on Taguchi L_9 orthogonal design is shown in Table 2.

Based on Table 1 Taguchi orthogonal design is formed using Minitab 16 software. The Taguchi orthogonal design and the experimental data are shown in Table 2. The experiment is conducted on the shown level of operating parameters to reduce the number of experiments [14, 15].

3 Optimization

In order to optimize the operating parameter the two methods are adopted.
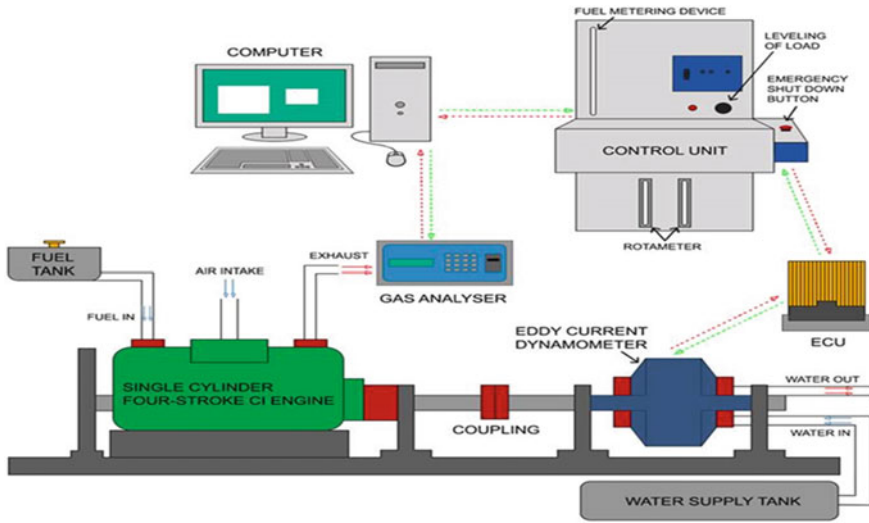
Fig. 1 Experimental setup

Table 2 Taguchi L₉ orthogonal design with experimental data for pure commercial diesel in single cylinder DICI VCR engine

Test No.	Number of levels			Experimental result			
	CR	IT	IP	Peak pressure (MPa)	RPR	NHR	MGT
1	12	10	30	42.03	3.12	38.01	1780.88
2	12	15	35	41.93	3.86	39.56	1888.56
3	12	20	40	42.09	3.56	38.89	1790.66
4	14	10	35	38.26	1.53	39.56	2016.44
5	14	15	40	38.88	1.79	35.70	1794.59
6	14	20	30	38.19	1.51	35.98	1804.86
7	18	10	40	47.35	3.24	34.37	1494.70
8	18	15	30	47.67	3.15	33.95	1464.46
9	18	20	35	47.22	3.19	33.66	1480.67

(i) All combustion parameter is taken as the single objective function individually and

(ii) Using an algorithm grey relation analysis, all the operating parameters are taken as objective function as a whole.

For the both of the two method the set of operating parameter is given in which the combustion parameters is optimized.

3.1 Method 1: All Combustion Parameter is Taken as the Single Objective Function Individually

If all the parameters is taken as a single objective function the following results is obtained. The Minitab 16 software is used to solve this L_9 orthogonal design. The signal to noise ratio (S/N ratio) is determine in order to find out optimal set of operating parameters. The S/N ratio is defined as the ration of signal power to the noise power. The higher the S/N ratio is the favourable conditions and accordingly will give the optimal set of operating parameter. The main effect plots for S/N ratio is shown in Fig. 2 for all combustion parameters.

All the four combustion parameters are taken as larger is better criteria. The peak pressure can be obtained by set of operating parameters CR3IT2IP3 (compression ratio 18, injection timing 15° before TDC and 40 MPa injection pressure), rate of pressure rise (RPR) is higher by a set of operating parameters CR1IT2IP3 (compression ratio 12, Injection timing 15° before TDC and 40 MPa injection pressure), net heat release (NHR) is higher by a set of operating parameters CR1IT1IP2 (compression ratio 12, injection timing 10° before TDC and 35 MPa injection pressure) and mean gas temperature (MGT) is higher by the set of operating parameters CR2IT1IP2 (compression ratio 14, injection timing 10° before TDC and 35 MPa injection pressure).
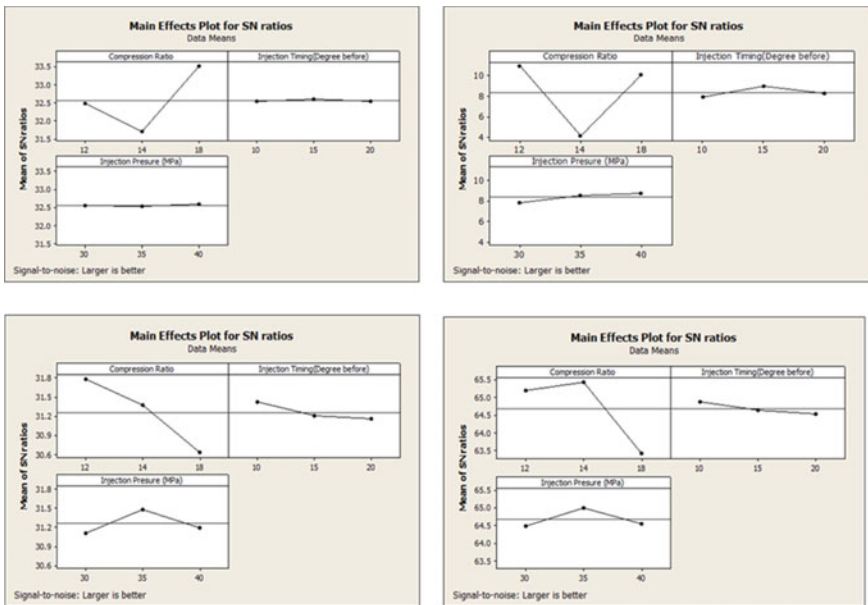
Fig. 2 S/N ratio plots of peak pressure, RPR, NHR and MGT, respectively

3.2 Method 2: Using an Algorithm Grey Relation Analysis, All the Operating Parameters Are Taken as Objective Function as a Whole

For multi objective optimization using grey relation analysis the first step is to normalize the experimental data using the Eq. 1. The normalized data is shown in Table 3

$$x_i^*(k) = \frac{x_i(k) - \min x_i(k)}{\max x_i(k) - \min x_i(k)} \tag{1}$$

In the second step deviation coefficient to need to determine which in given by Eq. 2. The deviation coefficient is shown in Table 4

Table 3 Normalized experimental data

Normalized experimental data				
Test No.	Peak pressure (MPa)	RPR	NHR	MGT
1	0.405063291	0.685106	0.737288	0.573245
2	0.394514768	1	1	0.768325
3	0.411392405	0.87234	0.886441	0.590963
4	0.007383966	0.008511	1	1
5	0.07278481	0.119149	0.345763	0.598083
6	0	0	0.39322	0.616689
7	0.966244726	0.73617	0.120339	0.054785
8	1	0.697872	0.049153	0
9	0.952531646	0.714894	0	0.029367

Table 4 Deviation coefficient

Deviation coefficient				
Test No.	Peak pressure (MPa)	RPR	NHR	MGT
1	0.5949	0.314	0.262	0.4267
2	0.605	0	0	0.231
3	0.588	0.127	0.1135	0.409
4	0.992	0.9914	0	0
5	0.9272	0.8808	0.654	0.4019
6	1	1	0.606	0.3833
7	0.0337	0.263	0.879	0.9452
8	0	0.3021	0.950	1
9	0.04	0.2851	1	0.9706

$$\Delta_{0i}(k) = \left| x_0^*(k) - x_i^*(k) \right| \tag{2}$$

The subsequent step is to find out the grey relation coefficient which is given by Eq. 3, the grey relation coefficient is given in Table 5

$$\xi_i(k) = \frac{\Delta_{\min} + \zeta \Delta_{\max}}{\Delta_{0i}(k) + \zeta \Delta_{\max}} \tag{3}$$

$\zeta = 0.5$, given all parameter are given equal preference.

After computation of grey relation coefficient the next step is to determine the grey relation grade which is give by Eq. 4. The corresponding grey relation grade and order is give in Table 6

$$\gamma_i = \frac{1}{n} \sum_{k=1}^{n} \xi_i(k) \tag{4}$$

Table 5 Grey relation coefficient

Grey relation coefficient				
Test No.	Peak pressure (MPa)	RPR	NHR	MGT
1	0.456	0.613	0.655	0.539
2	0.4522	1	1	0.6833
3	0.4592	0.7966	0.8149	0.550
4	0.3349	0.3352	1	1
5	0.3503	0.3620	0.43318	0.5543
6	0.333	0.333	0.451	0.566
7	0.9367	0.654	0.3624	0.345
8	1	0.6233	0.344	0.333
9	0.913	0.636	0.333	0.3399

Table 6 Grey relation grade and order

Test No.	GRG	Order
1	0.5663	6
2	0.7839	1
3	0.655	3
4	0.6675	2
5	0.4249	8
6	0.4211	9
7	0.574	5
8	0.575	4
9	0.555	7

Table 7 Response table for mean GRG

Operating parameter	Grey relation grade			Main effect (max–min)	Rank
	Level 1	Level 2	Level 3		
CR	0.6685*	0.5046	0.5687	0.1639	1
IT	0.6029*	0.5947	0.5441	0.0589	3
IP	0.5209	0.6691*	0.5517	0.1482	2
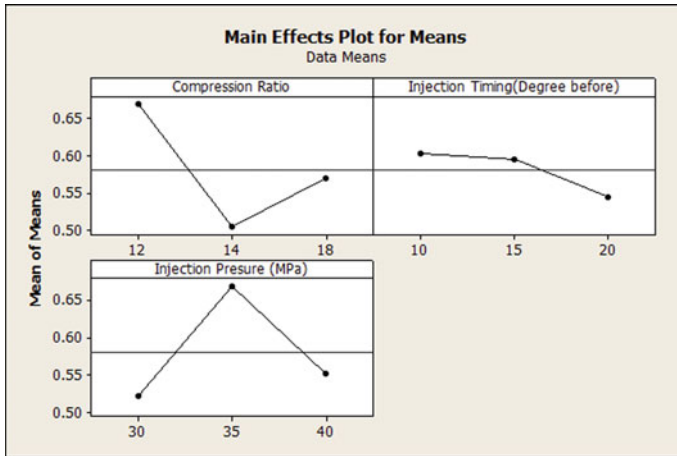
Fig. 3 Mean effect plot of GRG

As the DOE is orthogonal, then it is possible to separate the effect of each operating parameter on the grey relation grade at different level Table 7. The larger the GRG shows that better the combustion for a particular operating parameter. Table shows that CR1IT1IP2 (compression ratio 12, injection timing 10° before TDC and injection pressure 35 MPa) shown the optimal condition on which the combustion is improved. Figure 3 shown the mean effect plot for GRG.

3.3 Verification of Optimal Operating Parameter Through Confirmation Test

The experiment is conducted on the optimal condition of operating parameter that we obtained from the multi objective optimization method and initial process parameter. The value obtained from the experiment is shown in Table 8.

From the above confirmation experiment it may conclude that the optimal condition gives the improve combustion characteristics compare to initial operating parameter.

Table 8 Results of combustion parameter using the initial and optimal process parameters

| | Initial process parameter | Optimal process parameters |
		Prediction/experiment
Combustion parameter	CR2-IT2-IP2	CR1-IT1-IP2
Peak pressure (MPa)	38.44	41.22
RPR	1.71	3.08
NHR (kJ/s)	36.5	39.22
MGT (°C)	1801	1855

4   Conclusion

In this study, a multi objective optimization technique is used to predict the set of optimal operating parameters which will improve the combustion characteristics of variable compression ratio of compression ignition engine. It has been found that compression ratio 12, injection timing 10° before TDC and injection pressure 35 MPa will give the best combustion characteristics. If the user has the single objective such as peak pressure, RPR, NGT or MGT the optimal operating parameters are compression ratio 18, injection timing 15° before TDC and 40 MPa injection pressure, compression ratio 12, injection timing 15° before TDC and 40 MPa injection pressure, compression ratio 12, injection timing 10° before TDC and 35 MPa injection pressure, compression ratio 14, injection timing 10° before TDC and 35 MPa injection pressure, respectively.

References

1. Emiroğlu AO (2019) Effect of fuel injection pressure on the characteristics of single cylinder diesel engine powered by butanol-diesel blend. Fuel 256:115928
2. Asadi A, Kadijani ON, Doranehgard MH, Bozorg MV, Xiong Q, Shadloo MS et al. (2020) Numerical study on the application of biodiesel and bioethanol
3. Odeyale SO, Egunleti AO (2015) Prediction of fuel properties of biodiesel using two-layer artificial neural network. Walailak J Sci Technol 12(4):325–341
4. Frizziero L, Piancastelli L (2015) Accelerated FEM analysis for critical engine components. Walailak J Sci Technol 12(2):151–165
5. Wilson V, Udayakumar (2012) Optimization of diesel engine parameters using taguchi method and design of evolution. J Braz Soc Mech Sci Eng 34:423–428
6. Patel CM, Patel TM, Patel PR, Rathod GR (2019) Parametric optimization of single cylinder diesel engine for specific fuel consumption using plastic pyrolysis oil as a blend. Indian J Appl Res 236–238

7. Natraj M, Dhandapani N, Arunachalam VP (2005) Optimizing diesel engine parameter for low emissions using Taguchi method: variation risk analysis approach-part I. Indian J Eng Mater Sci 169–181
8. Dhandapani N, Nataraj M, Arunachalam VP (2005) Optimizing diesel engine parameter for emission reduction using Taguchi method: variation risk analysis approach-part II. Indian J Eng Mater Sci 505–514
9. Prabhahar M, Prakash S, George I, Amith KK, Swaroop (2020) Optimization of performance and emission characteristics of bio diesel fuelled VCR engine using Taguchi approach. Mater Today Proc
10. Shailaja M, Raju S (2016) Optimization of diesel engine parameter for performance, combustion and emission parameters using Taguchi and grey relational analysis. Global J Res Eng
11. Agarwal A, Dhar A (2013) Effect of fuel injection timing and pressure on combination, emissions and performance characteristics of a single cylinder diesel engine. Fuel
12. Satyanarayana K, Padala VK, Rao TVH, Umamaheswararao SV (2015) Variable compression ratio diesel engine performance analysis. Int J Eng Trends Technol (IJETT) 28(1):6–12
13. Mishra PC, Kar S, Mishra H, Gupta A (2016) Modeling for combined effect of muffler geometry modification and blended fuel use on exhaust performance of a four stroke engine: a computational fluid dynamics approach. Appl Therm Eng 89:1105–1118. https://doi.org/10.1016/j.applthermaleng.2016.08.009
14. Gupta A, Mishra PC (2018) Optimisation of emission characteristics of petrol engine running on alternative fuel and fitted with chambered type muffler: combined CFD and experimental method. Oxid Commun 41(1):11–22
15. Gupta A, Mishra PC (2018) Emission and friction analysis of IC engine running in methanol blend. Tribol Ind 40(1):10–18

Odia Handwritten Characters Recognition Through Cost–Benefit Analysis

Anupama Sahu, Sarojananda Mishra, Aditya Agrawalla, and Priyam Pradhan

Abstract Optical character recognition (OCR) document may be in handwritten or machine-printed script that can be used as input for a system to convert into an digital text format with read/write operation. OCR documents for different language scripts are an emerging concepts of today research. Particularly, in Indian languages, it is highly required in the current research era. This is an attempt toward making an OCR system for the Indian regional language such as Odia script. Local language such as Odia has great challenges to design the OCR character due to the big number and form of character within the Odia alphabet, the delicate ways within which they combine and lots of characters being roundish and similar in looks. In this paper, we present the classification approaches for Odia character recognition with cost–benefit analysis. Hence, our research work was focused toward analysis of a machine learning algorithm to classify the offline Odia handwritten character recognition through decision table and Naive Bayes in WEKA environment with cost–benefit analysis.

Keywords OCR · Odia character · Handwritten · Cost–benefit analysis · Machine learning

1 Introduction

Optical character recognition, often referred to as OCR, is the translation of typewritten or handwritten text into machine-readable and editable form. Some employment potentials of OCRs are as follows: preserving old and historical things in documents format in electronic mode, desktop publication, reading aid for the blind, ledgering, library cataloging, automatic reading by the machine for sorting the postal communication, bank check, and other handwritten and printed documents, etc. More efficient OCR packages are commercially available to perform such job in different languages such as English, Chinese, and Japanese. Typewritten or printed documents

A. Sahu (✉) · S. Mishra · A. Agrawalla · P. Pradhan
Department of Computer Science Engineering and Applications, Indira Gandhi Institute of Technology, Sarang, Dhenkanal, Odisha, India
e-mail: anupamasahu.21@gmail.com

can be processed by these systems. The characters can be recognized with different font style and sizes including intermixed text with graphics. While an outsized the amount of literature is getting for recognizing the Chinese, Roman, and Japanese characters, relatively less work is reported for recognizing the Indian language scripts such as Telugu, Odia, and Marathi. Thus, an urgent need to develop an economically efficient OCR method for Odia language was felt.

Moreover, there are many ancient texts, handwritten manuscripts, and accounts from the history of mankind that are written in different ancient and many modern indigenous languages which are easily perishable. Even after us, being in the so-called modern twenty first century is sometimes not able to preserve this data. It is not because we don't have technical infrastructure, we have devices and technology to preserve the scripts, but the problem is that the technical infrastructure we have is not smart enough to understand those texts and store them. Thus, the aim of this project is to find a way to advance our technological capability to understand and store the ancient texts for future references.

The rest of the paper starts with Background in Sect. 2. Section 3 described Proposed Model followed by discusses the simulation in Sect. 4. Result discussion of proposed work is followed by conclusion and future study and lastly references.

2 Background

Authors in [1] has presented the work on OCR on characters of Korean and Chinese for line-level transcriptions. However, in recognizing Chinese and Korean language, one major drawback that they encountered is that there are thousands of unique characters for recognition. They encountered that there are many unique characters which do not have enough training examples even though the numbers of training text are large. Hence, to solve this problem, they found that decomposing the characters into smaller fragments which are common to many characters may reduce the size of training models and will also allow the training examples to be shared to different characters with similar smaller fragments. In [2], it has been illustrated a combination of two different feature extraction techniques, namely HOG and Gabor filter, also have they, proposed a feed-forward neural network (Extreme learning machine—ELM) to calculate the perceiving rate of input characters. Their proposed approach of sum of vectors (feature) has been brought out a trademark recognition output of 96.1% where individual feature extraction methods have produced low accuracy output in comparison. So, deep learning (based on neural network) was introduced as it imitates human behavior and because of its good accuracy of 78.6% and the desired time.

Authors in [3] has used the approach to work on OCR on Gujarati characters using a strategy called design template matching technique. Inaccurate segmentation of a line, word, or character reduces the recognition accuracy. In addition to a variety of writing styles of different writers, the highly curved shapes of the characters also make the segmentation of Gujarati script in particular a difficult task to do. To solve

these problems of errors, radial basis probabilistic neural network (RBPNN) method was used due to its high noise tolerance. Some researchers found the recognition accuracy to be 91.16%, and others found the success rate of this approach to be 63%. Abdalkafor [4] has illustrated the work on OCR on Arabic characters otherwise known as ACR. The current researches in the fields of Arabic character recognition have success in few due to the complex nature of said language and scripts, and also, the characters take three other different forms. They found that the only solution was to use different databases to override the complex nature in the Arabic character to improve the low recognition accuracy.

Prameela [5] has proposed character recognition of Telugu characters in three stages which are preprocessing, feature extraction, and classification. In preprocessing, phase filtering of images is performed using median filters by them. Then, for evaluating the uniqueness of each character and to obtain the feature vector values, they have placed the image into a 3 * 3 grid to skeletonize the image of handwritten characters and have evaluated the corresponding centroid for all the nine zones. Gonzalez [6] has used MATLAB as a tool for implementing the algorithms, and the neural network is designed, but still, finding the correct model of neural network trial and error method is applied. To remove the noise, the image was converted into gray scale and then into binary image. Lines and then characters are extracted from the extracted line images. Verma [7] here has proposed a system to identify a diverse font of characters which will help in improving the OCR system accuracy. They exhibited a procedure for increasing the recognition accuracy of Devanagari optical character recognition system by creating an idea for discovery of emphasized words such as italic, bold, and underlined words.

Anupama [8] here has presented two machine learning algorithms, i.e., Naïve Bayes and decision table for identification of offline Odia handwritten character recognition. Anupama [9] here has proposed two novel algorithms, i.e., Naïve Bayes and decision table to classify the offline Odia handwritten character recognition. Initially, they have collected 240 samples of distinct offline Odia handwritten characters (without noise) of 20 subjects. After comparison of results on the basis of these two machine learning techniques, a decision table has been designed, and it has been found that the maximum accuracy is about 86% then Naïve Bayes. This paper is the extension work in [8, 9], which express the enhancement of accuracy through cost–benefit analysis.

3 Proposed Model

The cost–benefit analysis is referred to as the mechanism which is used to compute the benefits of a taking action against the costs involved with the action. Speaking in simple language in this paper, the potential class label from a situation is summed, and then, the total costs associated with taking that action are subtracted.

Data processing and implementation of different machine learning approaches and visualization tools, all these which help us develop a proper machine learning
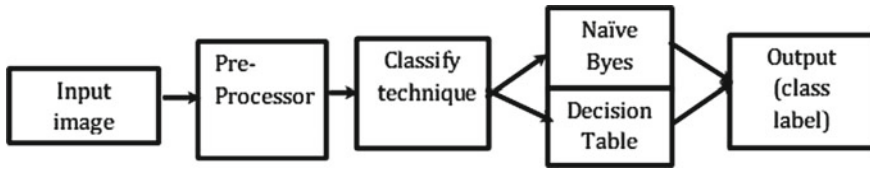
Fig. 1 Architecture to classify

method in order to apply the same to real-world data, are all provided by the open-source software named WEKA. Data preprocessing, classification, association rules, clustering, regression, and visualization are all the important and essential tools that WEKA accommodates. This is the reason why WEKA was used to channel this paper toward developing two approaches to classify the offline Odia handwritten character recognition. The Odia character images can be easily classified using these two approaches for recognition after preprocessing as in Fig. 1.

Here, preprocessing is the step where the input images are formatted before they are used by model training This may include resizing, orienting, and color corrections. 960 pictures are collected from 20 subjects (different writers) along with a fixed size of 60×60, in the image acquisition step which is extended work of previous works [9]. All the pictures are stored in a single folder. The data processing tools use the pictures using the file (eith extension of .arff), provided by WEKA. Sound has been labeled and eliminated in the preprocessing process. Then, two ML algorithms, namely Naive Bayes and decision table, are applied in the classification step.

4 Simulation and Result Discussion

In this section, decision table and Naive Bayes algorithms have been applied in WEKA environment. In this work, four characters from Swara Barna and four from Byanjan Barna of Odia language have been introduced only for Naive Bayes and decision table. Initially, 960 samples have been gathered from distinct Odia hand-written characters of 20 subjects (different writers). The decision table and Naive Bayes approaches have been executed to classify the Odia handwritten characters. The accuracy observed in those works is enhanced from 85% to different enhanced values for different characters such as Ka, Kha, Cha, and Chha. The enhanced accuracy varied between 99.14 and 99.78%, for example, 99.78% for CHHA, 99.57% for Ee, and so on.

For Decision Table

It is seen from Fig. 2 that the cost has been minimized to 0, which results the benefit is 39.15 by taking sample size 18.33%. It has been also observed that the classification accuracy has increased from 85 to 100%. From Fig. 3, it has been observed that the accuracy is more as the ROC area is closer to one.
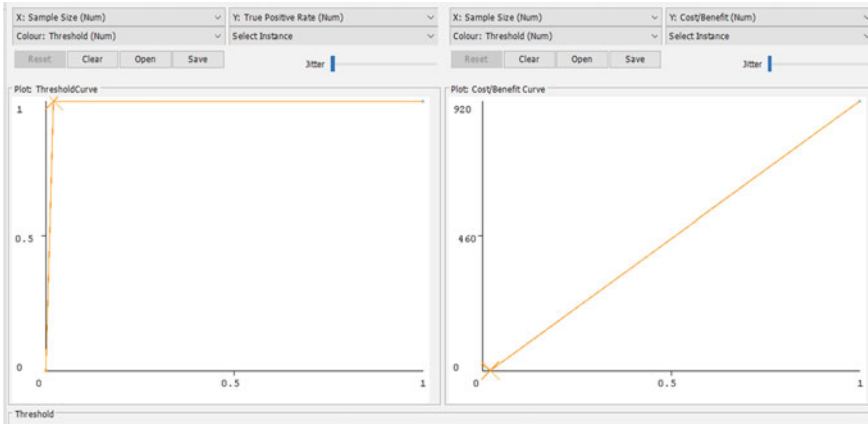
Fig. 2 Threshold and cost–benefit curve on decision table model for A

Fig. 3 ROC curve for
A using decision table
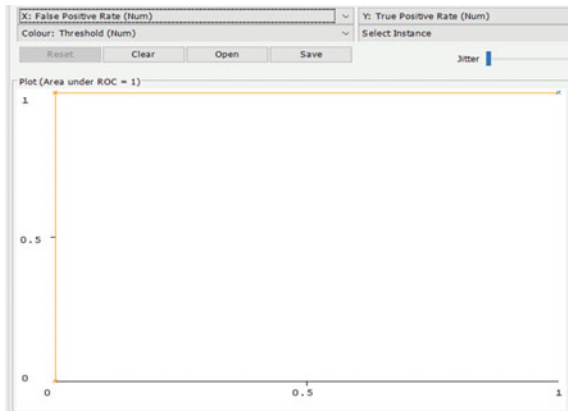
Similarly, for cha in Fig. 4, the cost has been minimized to 0, which results the benefit is 39.15 by taking sample size 18.33%. It has been also observed that the classification accuracy has increased from 85 to 100%. Here, from Fig. 5, it has been observed that the accuracy is more as the ROC area is closer to one.

The cost–benefit curve and ROC curve have also been analyzed for Aa, Chha, E, EE, Ka, Kha, and others.

By using sample size of 18.33% for each character, the cost will be minimized to enhance the old 85% accuracy level to 100% as observed in Table-1 for Aa, 99.57% for Aa, 99.78% for Ee, 100% for ka, 99.89% for Kha, and 99.78% for chha.

For Naive Bayes

It is seen from Fig. 6 that the cost has been minimized to 4, which results the benefit is 33.23 by taking sample size 18.33%. It has been also observed that the classification
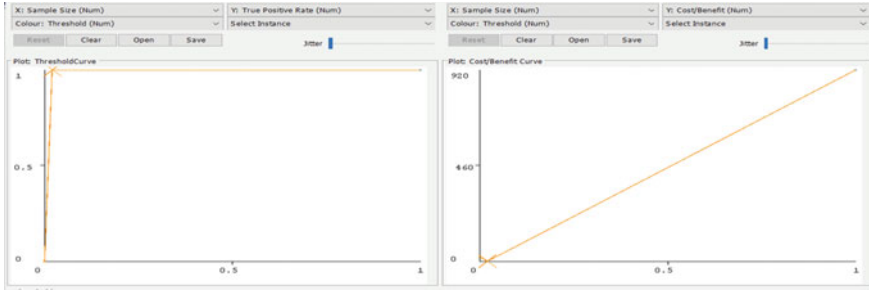
Fig. 4 Threshold and cost–benefit curve on decision table model for cha
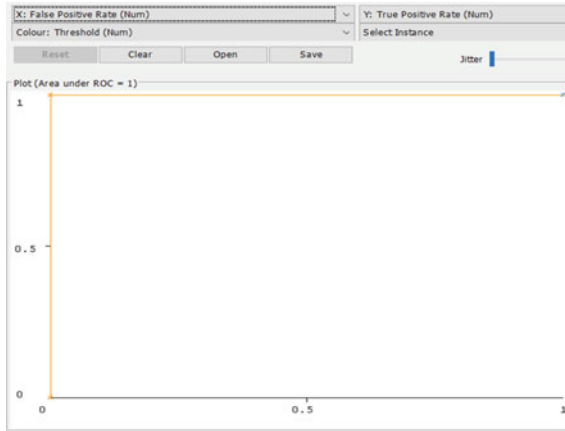
Fig. 5 ROC curve for cha using decision table

Table 1 New classification accuracies for different characters using decision table

Character (Swara Barna and Byanjan Barna)	Cost minimized to	Benefit	Sample size (in %)	Old accuracy (in %)	New accuracy (in %)
Aa	0	39.15	18.33	85	100
E	4	31.32	18.33	85	99.57
Ee	2	35.23	18.33	85	99.78
Ka	0	39.15	18.33	85	100
Kha	1	37.19	18.33	85	99.89
Chha	2	35.23	18.33	85	99.78
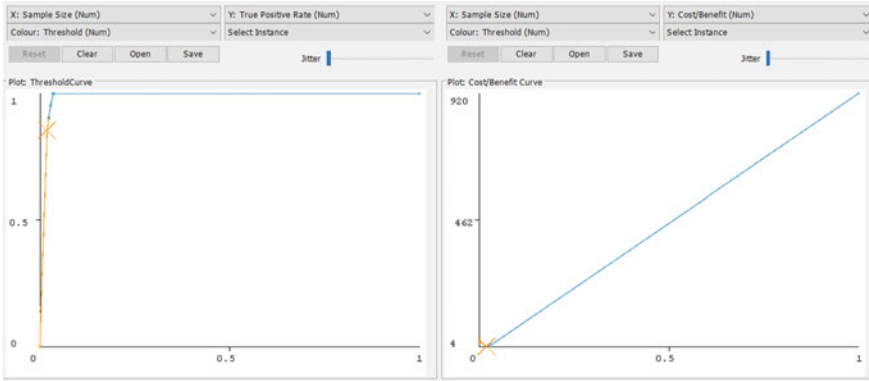
Fig. 6 Threshold and cost–benefit curve on Naive Bayes model for A

accuracy has increased from 85 to 99.57%. From Fig. 7, it has been observed that the accuracy is more as the ROC area is closer to 1.

It is seen from Fig. 8 that the cost has been minimized to 3, which results the benefit is 35.19 by taking sample size 18.33%. It has been also observed that the classification accuracy has increased from 85 to 99.68%. From Fig. 9, it has been observed that the accuracy is more as the ROC area is closer to 1.

By using sample size of 18.33% for each character, the cost will be minimized to enhance the old 85% accuracy level to 99.68% as observed in Table-2 for Aa, 99.36% for Aa, 99.78% for Ee, 99.14% for ka, 99.46% for Kha, and 99.57% for chha.

Fig. 7 ROC curve for A for Naive Bayes
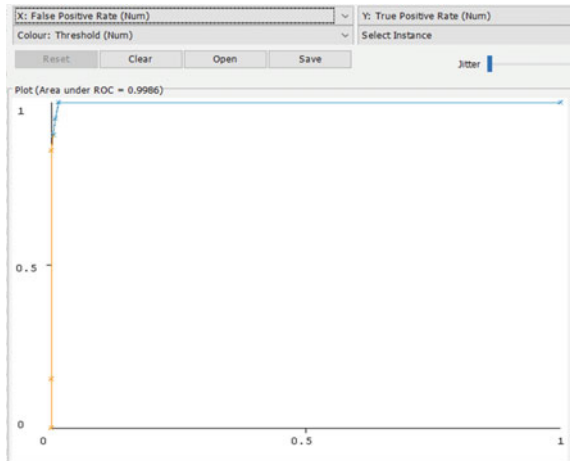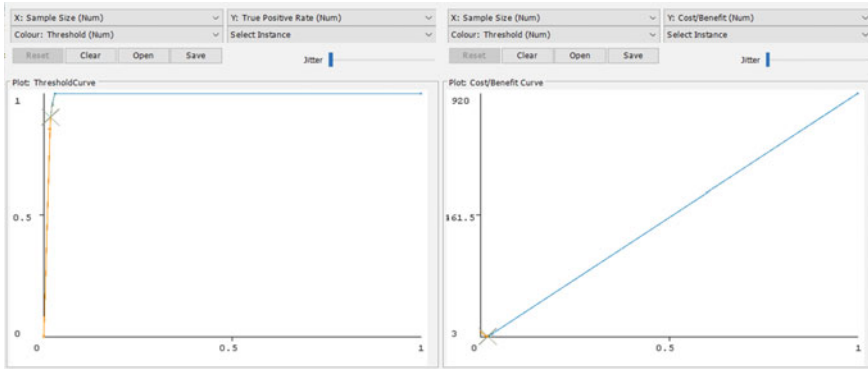
Fig. 8 Threshold and cost–benefit curve on Naive Bayes model for cha
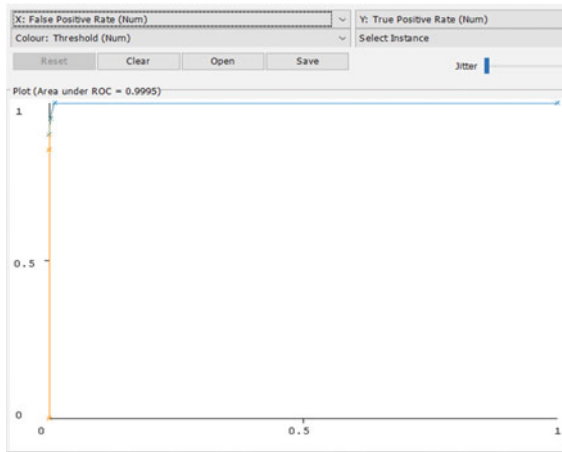
Fig. 9 ROC curve for cha using Naive Bayes

Table 2 New classification accuracies for different characters using Naive Bayes

Character (Swara Barna and Byanjan Barna)	Cost minimized to	Benefit	Sample size (in %)	Old accuracy (in %)	New accuracy (in %)
Aa	3	39.02	18.33	85	99.68
E	6	35.05	18.33	85	99.36
Ee	12	27.15	18.33	85	99.78
Ka	8	33.06	18.33	85	99.14
Kha	5	38.94	18.33	85	99.46
chha	4	37.06	18.33	85	99.57

5 Conclusion and Future Study

In our present work, we have demonstrated the effectiveness of different classification approaches for recognition of offline Odia handwritten characters through WEKA environment. We have taken 960 samples of distinct Odia handwritten characters from 20 subjects. The decision table obtained the maximum accuracy of about 86% in comparison to Naïve Bayes. Further, it has been extended through the conformation test through cost–benefit analysis. The enhanced accuracy after testing between the two classifiers is enhanced from 85 to 99.14%. Other classification algorithms can be considered for Odia Matra using large amounts of data in future.

References

1. Chang CC, Arora A, Perera LPG, Etter D, Povey D, Khudanpur S (2019) Optical character recognition with chinese and korean character decomposition. In: International Conference on Document Analysis and Recognition Workshop (ICDARW), pp 134–139
2. Bipu MMC, Afroge S (2018) A feature fusion based approach for handwritten bangla character recognition using extreme learning machine. In: International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2), pp 1–4. https://doi.org/10.1109/IC4ME2.2018.8465629
3. Dhara J, Risodkar Y (2018) Deep learning based gujarati handwritten character recognition. pp 563–566. https://doi.org/10.1109/ICACCT.2018.8529410
4. Abdalkafor AS (2018) Survey for databases on arabic off-line handwritten characters recognition system. In: 1st International Conference on Computer Applications & Information Security (ICCAIS), pp 1–6. https://doi.org/10.1109/CAIS.2018.8442001
5. Prameela N, Anjusha P, Karthik R (2017) Off-line telugu handwritten characters recognition using optical character recognition. In: International conference of Electronics, Communication and Aerospace Technology (ICECA), pp 223–226. https://doi.org/10.1109/ICECA.2017.8212801
6. Gonzalez RC, Woods RE, Eddins SL (2009) Digital image processing using MATLAB, fifth impression. Gatesmark Publishing, United States
7. Verma VK, Tiwari PK (2015) Removal of obstacles in devanagari script for efficient optical character recognition. In: International Conference on Computational Intelligence and Communication Networks (CICN), pp 433–436. https://doi.org/10.1109/CICN.2015.90
8. Sahu A, Mishra SN (2020) Odia handwritten character recognition with noise using machine learning. In: 2020 IEEE international symposium on sustainable energy, signal processing and cyber security (iSSSC), pp 1–4
9. Anupama S, Mishra SN (2021) Offline odia handwritten characters recognition using WEKA environment. In: International Conference on Machine Learning Internet of Things, and Bigdata, (Intelligent systems), Lecture notes in networks and systems, vol 185. Springer, Singapore

Image Encryption Using RSA Algorithm

Aradhana Sahoo, Pratyasha Mohanty, and Purna Chandra Sethi

Abstract In the current covid pandemic situation, secure online transmission of data has the highest precedence over other activities. For providing computational hardness that is for making tough to break the key for finding the unique message, there are various algorithms are present. For secure data transmission, many researchers have applied different cryptography algorithms and in order to improve the level of information security, different hybrid cryptography algorithms have been proposed. In cryptography algorithm implementation, key management plays a major role. For this reason, we have applied an image encryption technique in which a random image is considered as the key. Using the random image as a key, we have encrypted another image as information using the RSA algorithm. The comparison of the proposed method is done with the traditional approach and concluded that the cryptography algorithm implemented using an image as key provides more security in terms of encryption and decryption time.

Keywords Image encryption · Image as key · Performance enhancement · RSA algorithm

1   Introduction

Information is represented in various form such as text, image, audio, and video. The security and authentication are becoming major issue during data transmission over Internet. Hence, large-size key is used by various researchers during different cryptography algorithms implementation for increasing the level of data security. Though larger size key increases the level of data confidentiality, but the key management is becoming a tedious task. As an alternative to large-size key, image as key will lead to easier key management along with better information security.

Encryption technique is widely used for achieving information security. Encryption of image is the process that convert real image to a secure format with various

A. Sahoo · P. Mohanty · P. C. Sethi (✉)
Rama Devi Women's University, Bhubaneswar, Odisha 751022, India
e-mail: purna.sethi@gmail.com

encryption techniques. Consequently, it will be relatively difficult to access and corrupt the information without having the decryption key. RSA is one of the recognized public-key cryptography algorithm that facilitates the encryption and authentication of information.

In digital communication, encryption and decryption of image have various applications. The other fields like medical image processing and military message multimedia structures and telemedicine frequently using image security. The hospital, financial, administrative, defense-related fields, etc., are using images as the keys for achieving higher level of data security in comparison to information as key. Stealing the personal information is an unlawful obligation for any kind of institute. So, we have to make the encryption method much stronger for the image or data so that the data are transmitted securely, and it can't be chopped simply by the attacker. At the receiver side, the real image can find after decrypting it. At the receiver end, it can easily access the data or image by using the private key.

The research paper is structured over different sections. Section 2 of this research paper covers a brief insight in to the research work done by different researchers for cryptography algorithms implementation. Section 3 of this paper focuses on the motivation, and the proposed problem proposed research work followed by the proposed method and performance analysis in terms of experimental result is explained in Sect. 4. Finally, the conclusion and the future scope are specified in Sect. 5.

2 Literature Overview

Information security is the major concern of current online-based applications. While transmitting the data over online medium, it must be protected from unauthorized access. Many researchers proposed many security algorithms and many hybrid approaches for achieving information security [1–7]. Faster algorithms and high performance computing techniques lead to some mean for cracking of data even after security algorithm implementation. So, different researchers are motivated toward image security. Network traffic analysis involve management of various resources out of which bandwidth plays a major role. The researcher proposed a dynamic bandwidth management technique [8] for on-demand service. Since the images involve more bandwidth in comparison to text information, the proposed work could be integrated with dynamic resource management technique for faster and secure service.

These days mix media data have been moved speedily and extensively through the Web into different structures like picture, sound, video, and text. In computerized communication over the Web, each information is apparent and available to each client. So, security of data is an essential and significant task. While transmitting the data from one place to the other over Internet, it must be protected from illegal access. So, it is most important to transfer the image securely.

There are many methods to save pictures including encryption, watermarking, digital watermarking, adjustable watermarking, cryptanalysis, and steganography. This paper is based on image security using RSA cryptographic algorithm. RSA is

a common or frequently used algorithm which is accepted by many researchers for many applications.

2.1 RSA Algorithm

RSA algorithm is one of the frequently used asymmetric key cryptography algorithm that was developed by Ron Rivest, Adi Shamir, and Leonard Adleman in 1977. The naming was done based of the surnames of the developers. The algorithm can be used for achieving both security as well as authentication of the information.

Public-key cryptosystems are mostly used to secure the data during communication. RSA is an asymmetric key cryptography algorithm that uses public and a private key pair. Hence, the encryption of message are done at the sender end using the public key and the decryption are done at receiver end using the private key of receiver respectively.

The reason of using RSA algorithm is the major problem in factorization of large integers. The security occurs due to the product of two large prime numbers during the algorithm implementation. The most compound part of RSA cryptography is the generation of public key and private key. Using Rabin–Miller primality test algorithm, two prime numbers are generated that are p and q. The link between private key and public key is provided by the two prime numbers. The key size is frequently expressed in bits.

2.1.1 Steps of RSA Algorithm

RSA algorithm involves three major steps during implementation. They are as follows:

A. Generation of key
B. Encryption
C. Decryption.

A. **Key Generation**
First stage of RSA algorithm is the key generation that involves public and private generation. As the name represents, public key is an open key that can be seen to everyone, and it is used to participate in encryption of messages. Transmitting images are going through encryption using public key, and after that it can perform decryption via using private key. The keys of RSA algorithm can create using the subsequent steps,

1. First, we have to select the two different prime numbers that is p and q.
2. For safety, prime integers p and q should be selected with same bit-length. Prime integers are proficiently found by the primality testing.
3. Then, we have to calculate the value of n that is $n = pq$.
4. n is the modulus that is used for equally the public and private keys. Its length is known as key length that is usually stated in bits.

5. We have to compute Euler's totient of n. $\varphi(n) = \varphi(p)\varphi(q) = (p-1)(q-1) = n - (p+q-1)$; here, φ is Euler's totient function. This rate is kept private.

6. Then, we have to choose an integer e that $1 < e < \varphi(n)$ and $\gcd(e, \varphi(n)) = 1$; i.e., e and $\varphi(n)$ are coprime. e is out as key which kept public. e has a brief bit-length and slight Hamming weight outcomes in more effective encryption. However, much minor e values have been publicized to become less locked in some settings.

7. Determine d as $d \equiv e - 1 \pmod{\varphi(n)}$, i.e., d which is the modular multiplicative inverse of e (modulo $\varphi(n)$). This is performed as, solve d given by $d \cdot e \equiv 1 \pmod{\varphi(n)}$. That is calculated using extended Euclidean algorithm. It uses the pseudo-code in the modular integers section; inputs a and n correspond to e and $\varphi(n)$, respectively.

8. Evaluate the value of d which is kept as the private key.

Public key involves the modulus of n and e. The private key has the modulus of n and d, and it kept secret. p, q, and $\varphi(n)$ values are kept secret because these values can be used for calculating d.

B. **Encryption**

$$c \equiv m^e \pmod{n} \big[c = \text{cipher text}, m = \text{plain text}, e = \text{public key}, d = \text{private key} \big]$$

C. **Decryption**

$$m \equiv c^n \pmod{n} \big[c = \text{cipher text}, m = \text{plain text}, e = \text{public key}, d = \text{private key} \big]$$

This method is traditional one which is used for encryption and decryption using text or number as a key. Here, we proposed an algorithm same as the previous with a simple modification that we use the image as a key instead of text or number as key. The following is our proposed algorithm:

2.1.2 RSA Algorithm Illustration

Key Generation

1. Let 2 prime numbers $p = 3$, $q = 11$
2. Calculating n

$$n = p * q = 3 * 11 = 33$$
$$\Rightarrow n = 33$$

$\varphi(n) = (p-1) * (q-1)$
3. $\varphi(n) = (3-1) * (11-1)$
$\varphi(n) = 2 * 10 = 20$

4. Assume e such that $\gcd(e, \varphi(n)) = 1$ and $1 < e < \text{phi}(n)$
    Let, $e = 7$
5. Calculating d

$$\text{de mod } \varphi(n) = 1$$
$$\Rightarrow 7 * d \text{ mod } 20 = 1$$
$$\Rightarrow d = 3$$

    Since $e = 7$, $d = 3$
6. Public key $= \{e, n\} = \{7, 33\}$
7. Private key $= \{d, n\} = \{3, 33\}$
    Encryption Plaintext $M < n$ and let, $M = 31$

$$C = 31^7 \text{ mod } 33$$
$$C = 4$$

Decryption $M = 4^3 \text{ mod } 33$
$M = 31$.

2.2 Proposed Algorithm

In our proposed algorithm, we have considered image as information instead of text or number and another image as the key for the RSA algorithm implementation.

The steps of the proposed algorithm are as follows:

1. Consider the image and find out the array format.
2. Finding the length of an array.
3. Consider the key image and find out the corresponding array format.
4. From the key image, any random prime number will be considered as key (For simplicity, we have considered the first prime number as key among the key image array).
5. Then, the same key generation, encryption, decryption process can be done as the traditional one.

2.2.1 Key Generation

1. Choose two numbers p and q that are prime and distinct in nature.
2. For high security purposes, the prime numbers p and q should be randomly taken and must of same bit-length.
3. Compute $n = pq$, where n is used for modulus of both the public and private keys. Its size is expressed in bits which is known as key length.
4. Compute $\varphi(n) = \varphi(p)\varphi(q) = (p-1)(q-1) = n - (p+q-1)$, where φ is Euler's totient function.

5. Choose an integer e such that $1 < e < \varphi(n)$ and $\gcd(e, \varphi(n)) = 1$, i.e., e and $\varphi(n)$ are coprime.
6. Determine d as $d \equiv e^{-1} \pmod{\varphi(n)}$;
    i.e., d is the multiplicative inverse of e (modulo $\varphi(n)$).
    Solve d given $d \cdot e \equiv 1 \pmod{\varphi(n)}$.

2.2.2 Encryption

$$c \equiv m^e \pmod{n} \big[c = \text{cipher text}, m = \text{plain text}, e = \text{public key}, d = \text{private key} \big]$$

2.2.3 Decryption

$$m \equiv c^n \pmod{n} \big[c = \text{cipher text}, m = \text{plain text}, e = \text{public key}, d = \text{private key} \big]$$

3 Motivation and Problem Statement

To provide security to an image-base data or image, another image can be used as key instead of general key for protecting the data from illegal access. Here, we take another image as key so that during encryption process whatever result produce cannot be easily accessed by the attacker. Management of information during transmission is also easier in comparison to the normal test as a key.

Image encryption is a useful technique of image content protection. It is the procedure of encoding underground image with the help of RSA algorithm in such a way that illegal users can't contact it. RSA is known as asymmetric cryptographic algorithm. It uses two distinct keys for this process. From the two distinct key, one of which is using for encryption, and the other one is used for decryption of the image. So, we try to implement RSA algorithm for our image encryption using another image as a key prior to this the algorithm was used for image or file encryption using digit or string as a key. While using text as a key, there may be chances to loss some bit between transmissions of data, but when we consider the image as a key, it will more secure than the previous.

Image encryption method prepares information unreadable. Therefore, no hacker and other unauthorized users have access to original information through public network.

3.1 Image Security

In today's Internet, digital images are widely communicated. It is an essential and challenging task to provide security to digital images on shared communication

Encryption Flow

Image Input

⇩

Load Image

⇩

Array Conversion

⇩

RSA Algorithm (Encryption)

⇩

Generate Cipher text file

⇩

Stop

Decryption Flow

Cipher text file Input

⇩

Load file

⇩

RSA Algorithm (Decryption)

⇩

Array to Image Conversion

⇩

Write to File

⇩

Stop
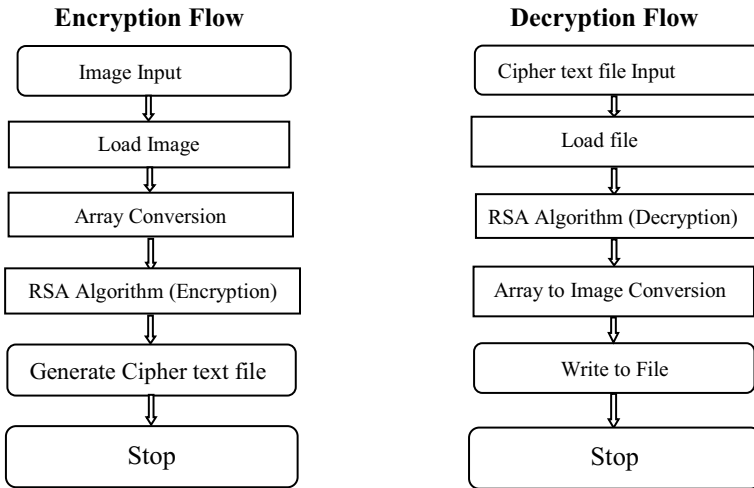
Fig. 1 Flow of encryption and decryption

channel. The transmission of image over the Internet is not secure, so to provide security and authentication to the images or data, we need various techniques for image security. We use encryption–decryption technique here for image security. To provide protection to the image, we use image encryption along with an image. Here, we have used RSA algorithm for our image encryption process.

We have considered an image as an information and another image as key. Both images are converted into an integer array for identification of information and key for the implementation of the algorithm. The encryption and decryption process is applied on the individual elements of array and the encrypted image as well as the original images are generated. For the performance of the algorithm, we have applied the proposed method four times on four distinct images, and the average time for encryption and decryption process is calculated.

4 Proposed Method and Performance Analysis

RSA algorithm uses discrete logarithm approach for implementation. Discrete we have used RSA algorithm in our research work for achieving image security. Since, RSA algorithm involves discrete logarithm approach and it is very difficult was used in many real life applications for information security, so we have used this algorithm was used for the security of information. The proposed image encryption technique is implemented by considering a random image as a key. The flow representation of the whole encryption and decryption process is depicted in Fig. 1.

4.1 Encryption

In encryption, the plain text content represented in the form of image is converted cypher text using another image as a secrete key. The picture can likewise be changed over to scrambled structure utilizing the random image as key and the resultant image after encryption is represented in Fig. 2. The scrambled picture is then sent over an insecure channel to the receiver. At the receiver end, the scrambled picture is decoded using the private key of the receiver. The proposed image encryption technique is used for providing better security of information. Subsequent, to encoding information, objective-scrambled information was decoded with assistance of association called as unscrambling. The resultant encrypted image are generated by applying the RSA algorithm between information image and the key image i.e. the binary equivalent of information image undergoes encryption with each pixel corresponding binary equivalent of key image in order to produce the encrypted image. The resultant encrypted image is depicted in Fig. 3.

Fig. 2 Original image

Fig. 3 Encrypted image
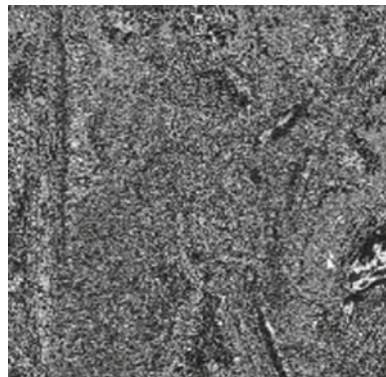
4.2 Decryption

The resultant encrypted image undergoes decryption using RSA algorithm. The resultant image after decryption is depicted in Fig. 4 which is same as that of the original image. In our research work, we have considered four different images out of three are gray scale image and one color image for encryption and decryption using RSA algorithm. The resultant encryption and decryption for each images using the traditional and the proposed approach are depicted in Fig. 5 and Fig. 6 respectively.

Fig. 4 Decrypted image

Original Image	Encrypted Image	Decrypted Image	Encryption Time	Decryption Time
			6.56 sec	5.66 sec
			2.81 sec	2.38 sec
			2.95 sec	2.57 sec
			14.6 sec	12.3 sec

Fig. 5 Resultant table for traditional approach (while using text/number as a key)

Fig. 6 Resultant table for proposed algorithm (while using image as a key)

Table 1 Experimental result using the proposed method

	Encryption time of traditional approach (s)	Decryption time of traditional approach (s)	Encryption time of proposed algorithm (s)	Decryption time of proposed algorithm (s)
Image1	6.56	5.66	10	7.34
Image2	2.81	2.38	4.38	3.36
Image3	2.95	2.57	4.12	3.16
Image4	14.6	12.3	21.4	16.2
Average time	6.73	5.727	9.975	7.515

The time required for different operations based on our proposed approach is represented in Table 1.

It shows the calculated time values for both the algorithms. If we compare the average time of this traditional approach and proposed algorithm, then we will see that the average time of encryption and decryption is high, while using image as a key. So, it is more secure than the traditional approach. That it will take more time to encode and decode the information by an attackers, so it is more secure (Fig. 7).

The calculated time for different activities during the algorithm implementation could be represented as:
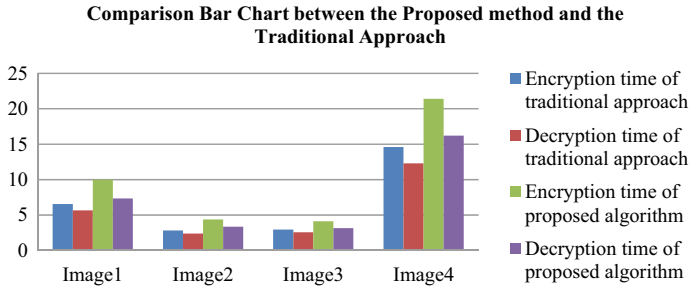
Fig. 7 Comparison bar chart

	Encryption time of traditional approach (in ms)	Decryption time of traditional approach (in ms)	Encryption time of proposed algorithm (in ms)	Decryption time of proposed algorithm (in ms)
Image1	6.56	5.66	10	7.34
Image2	2.81	2.38	4.38	3.36
Image3	2.95	2.57	4.12	3.16
Image4	14.6	12.3	21.4	16.2

It represents the graphical data and comparison between two algorithms. In this bar chat, blue color bar shows the encryption time of traditional approach; red color bar shows the decryption time of traditional approach; green color bar shows encryption time of proposed algorithm, and navy blue color bar shows decryption time of our proposed algorithm. It shows that for every image, the time of encryption and decryption of proposed algorithm is higher than the traditional approach, so it will be more secured.

5 Conclusion

In the computerized world, the security of pictures has gotten more significant as the correspondence has expanded quickly. Every one of the methods is in an ongoing picture encryption that could just track down to a lowest degree of safety. Here, the picture that calculate encryption proposed productive and exceptionally securable with significant degree of safety and less calculation. The consequences of the recreation show that the RSA calculation enjoys benefits dependent on their methods which are applied on pictures. Henceforth, it is assumed that for picture encryption, the strategies are useful and give security in the open organization. In this digital world, as the communication increased rapidly, the security of image has become more important. It helps to share files with high security. Lost or stolen device can be protected using encryption.

- All the techniques we used are in real-time image encryption that could find a low level of security.
- Here, encryption using image as a key proposed highly secure with less computation.
- Result of this research show that the algorithm gives advantages based on several techniques which are applied on data images.
- There is no loss of data in the process as the decrypted image is the accurate of the original one.
- The performance analysis of the proposed algorithm is increased by 40% as compared to text as a key.
- Without knowing the private key, no one can determine the original image.
- Hence, it was concluded that various methods used here are good for encryption of image using image as a key, and it gives higher security in open network.

References

1. Forouzan BA. Cryptography & network security, 1st edn. TMH
2. Kahate A. Cryptography and network security, 2nd edn. TMH
3. Al Sabti KDM, Hashim HR (2016) A new approach for image encryption in the modified RSA cryptosystem using MATLAB. Glob J Pure Appl Math 12(4). ISSN 0973-1768
4. Sethi PC, Behera PK (2015) Methods of network security and improving the quality of service—a survey. Int J Adv Res Comput Sci Softw Eng 5(7):1098–1106
5. Sethi PC, Behera PK (2016) RSA cryptography algorithm using linear congruence class. Int J Adv Res 4(5):1335–1347
6. Sheu T-F, Huang N-F, Lee H-P (2010) In-depth packet inspection using a hierarchical pattern matching algorithm. IEEE Trans Dependable Secure Comput 7(2)
7. Durairaj M, Muthuramalingam K (2018) A new authentication scheme with elliptical curve cryptography for internet of things (IoT) environments. Int J Eng Technol 7(2.26):119–124. https://doi.org/10.14419/ijet.v7i2.26.14364
8. Sethi PC, Behera PK (2017) Network traffic management using dynamic bandwidth on demand. Int J Comp Sci Inf Secur (IJCSIS) 15(6):369–375

Identification of Schizophrenic Individuals Using Activity Records Through Visualization of Recurrent Networks

Rohit Kumar Bondugula, Kaushik Bhargav Sivangi, and Siba K. Udgata

Abstract Schizophrenia is a chronic brain disorder that affects the mental well-being of a person. The symptoms of active schizophrenia include hallucinations, psychosis, sedentary lifestyle, and disorganized speech, which are considered significant indicators, and if left undiagnosed, it results in severe damage to behavioral abilities. With the introduction of sensor-based data from smart wearable devices, we can track motor activities and use the data to identify such health states and help identify schizophrenia for potential diagnosis. The main objective of this work is to differentiate subjects suffering from schizophrenia based on the measured activity through sensors over a period of time. For our experiments, we employed a publicly available dataset containing actigraphy data of patients who have schizophrenia, as well as healthy individuals. We propose a bidirectional recurrent network to extract the relevant features from the activity signals to achieve this goal and achieved an accuracy of 86.60%. To identify the potential features that affect determining whether an individual is suffering from the disease, we applied an attention mechanism to identify the active parts of the activity signal and interpret the result. Experimental results indicate that the bidirectional recurrent network model can extract relevant features and identify the most relevant parts of the input signal responsible for schizophrenia.

Keywords Schizophrenia · Healthcare · Recurrent neural networks · Artificial intelligence · Attention mechanism

1 Introduction

Schizophrenia is a serious mental illness that has a detrimental impact on the patient's quality of life. Schizophrenia is one of the top ten causes of disability worldwide [1]. The illness is chronic and recurrent, with a wide range of disease load and degree of

R. K. Bondugula · K. B. Sivangi · S. K. Udgata (✉)
School of Computer and Information Sciences, University of Hyderabad, Hyderabad, India
e-mail: udgata@uohyd.ac.in

R. K. Bondugula
e-mail: rohitkumarbondugula@gmail.com

impairment among patients. A variety of symptoms may occur, including "positive" symptoms such as delusions, or psycho-motoric agitation and hallucinations, "negative" symptoms such as expression and diminished motivation or impaired affective experience and cognitive symptoms such as difficulties focusing, lack of problem solving and paying attention [2]. Antipsychotic medicine is the primary therapy for schizophrenia, for both the acute psychotic episodes and also for relapse prevention [1]. The therapeutic effectiveness of antipsychotics, as well as their negative effects, vary greatly among people. The use of subjective methods such as self-reports, clinical evaluations, and observations is crucial in the diagnosis of schizophrenia [3].

Sensor data gathered from motor activity recordings, on the other hand, may be an efficient tool that could either substitute or leverage existing subjective diagnostic methods [4]. Wrist-worn piezoelectric accelerometers, which measure motions in three dimensions, are widely used to track motor activity. When using various machine learning approaches with similar data, previous research has been able to successfully identify between depressed patients and controls [5].

Actigraphy is a noninvasive process of measuring human resting and activity cycles that is often monitored with a wrist-worn sensor to measure gravity acceleration units [6]. Data from actigraphs have been used in research of sleep [7], psychiatric diagnoses such as ADHD [8] and bipolar disorder [9] and to a lesser extent, schizophrenia research. A recent study summarized motor activity investigations of schizophrenia, all of which used standard statistical methods [10].

Schizophrenia has been found as a separate subtype in motor activity in a review contrasting the motor activity of schizophrenic subjects to mood disorder subjects and healthy controls using nonlinear mathematical models. Complex and irregular activity patterns [11] as well as a different profile in terms of the distribution of active and quiet phases [12], characterize this species. In addition, schizophrenia patients were linked to lower mean motor activity, irregular activity patterns, and poor sleep quality in a recent comprehensive study [10].

Sleep disruption is a prevalent symptom of a variety of mental diseases, and it is linked to a disruption in circadian rhythmicity [13]. The circadian rhythm is a biological pulse that is controlled by a brain-based internal clock. Humans are synchronized to the diurnal cycle of day and night by the clock, which controls a complex system of recurrent interlocking biological rhythms such as shorter rest–activity patterns, the sleep–wake cycle, hormone regulation, and a variety of other internal activities [14].

Even though the brain clock is the primary controller of this inner system, extrinsic influences such as social relation and habits, traumatic existential experiences, and then the vulnerability to artificial light all have an impact on and have the ability to disrupt this complicated vital system [15]. Sleep disruption is a typical sign of a disrupted and unsynchronized circadian clock in schizophrenia [16]. Both daily oscillations of the circadian clock and patterns of social rhythms are identified in motor activity.

Boeker et al. [17] proposed Hidden Markov model for the classification of schizophrenia and non-schizophrenic behavior over a set period of time. HMM application identifies active and non-active subjects trains the model. The obtained

result shows better performance compared with the existing methods to derive the features in respect to the goodness-of-fit and better performance in classification. Reinertsen et al. [18] worked on schizophrenia continuous assessment using heart rate and accelerometer data where they monitor schizophrenia changes over associated with changes in heart rate and physical activity measures. They used support vector machine model to classify the schizophrenia based on data obtained from human heart rate and physical activities over a period of 2–8 days. Experiment results show continuous tracking subjects of schizophrenia and estimate the illness severity. Jakobsen et al. [19] analyzed Schizophrenia data, which is collected from sensors. In their experiments they observed 22 schizophrenia patients and 32 healthy persons and give assess to the the patients.

Afonso et al. [20] examines differences between schizophrenia patient and healthy person controls in terms of quality of sleep patterns, quality of life (QoL). Pittsburgh Sleep Quality Index (PSQI) is used to measure the quality of sleep. World Health Organization Quality of Life—Abbreviated version (WHOQOL-Bref) evaluates QoL. They perform the examination on different patients, from their observations they concluded that Schizophrenia patients have low sleep quality, more disturbed sleep–wake patterns. Bondugula et al. [21] worked on a consensus model for the early classification. The goal of the work was to improve accuracy of prediction by implementing a weighted consensus model.

Reinertsen et al. [22] discussed how heart rate changes show impact on behavior, physiology and mood. They analyzed interactions between heart rate and locomotor activity and classifies the schizophrenia patients and non-psychiatric patients from controls. Osipov et al. [23] identifies metal disorders based on patients physical activities. This can be identified by on self-reports or observations. They discover feature selection based on physiological and physical activity data. To get these feature selection they used dynamics of Multiscale Entropy change over scales of HR signal, the mean HR, the standard deviation and mode of locomotor activity.

Electroencephalography (EEG), which measures electrical activity in the brain, is another approach for detecting and classifying schizophrenia [24]. In such data, machine learning shows promise for distinguishing between schizophrenia patients and healthy subjects. Nonetheless, data collection with electrodes put on the head appears to be a significantly more time-consuming and difficult operation than a simple wrist-worn actigraph capturing motor activity [25].

The main purpose of our research is to investigate the feasibility of using the activity data to classify and diagnose patients with schizophrenia. As a result, a dataset including activity recordings from people with schizophrenia disease and the control group is utilized to train a deep learning technique to determine whether or not a subject has schizophrenia.

1.1 Contributions to the Current Work

1. We extracted activity signals based on fixed time intervals. Concretely, we sampled 3 h interval signals during general active periods and used them to classify schizophrenic subjects.
2. We interpreted the learned features from the model through visualization of attention weights. This helps novices to understand the key areas of activity signal which are crucial in classification.
3. Existing works were done by inclusion of EEG as they result in more robust performance but it's expensive. We have experimented on activity data alone which is portable and can be worn without discomfort and the proposed model has reported commendable results on such data.

2 Dataset Description

2.1 Dataset

The dataset [19] included with this paper was recorded with a wrist-worn actigraph device (Actiwatch, Cambridge Neurotechnology Ltd, England, model AW4) with a piezoelectric accelerometer designed to record the integration of intensity, quantity, and duration of movement in the x, y, and z axes was used to capture motor activity. Movements greater than 0.05 g were recorded at a sampling frequency 32 Hz.

The dataset comprises actigraph data taken from 22 psychotic subjects treated at Haukeland University Hospital's long-term open psychiatric unit. All have schizophrenia and have tried antipsychotic drugs. There were 19 males and 3 females in the group. To diagnose the patients, the medical specialists performed a semi-structured interview based on DSM-IV criteria [26]. Seventeen of the patients were diagnosed as paranoid schizophrenic. The other five individuals did not have a specific subtype of schizophrenia, but they were non-paranoid. The current diagnostic handbook, DSM-5, does not distinguish schizophrenia subgroups [27]. The patients' current psychotic symptomatic state was assessed using the Brief Psychiatric Rating Scale (BPRS), a widely used rating scale for assessing the general psychopathology of schizophrenia patients. The BPRS is made up of eighteen components that are evaluated from one to seven, with higher cumulative scores indicating a more severe condition. For 1 min epochs, the output is an integer value proportional to the movement intensity.

Figure 1 represents the violin plots representing the distribution of activity values for both healthy and unhealthy classes. It is evident that the healthy subjects (class 0) show a higher mean value compared to schizophrenic subjects (class 1) which indicates lower motor activity of the patients suffering from schizophrenia.

Fig. 1 Violin plots depicting the range of activity values pertaining to healthy and unhealthy individuals
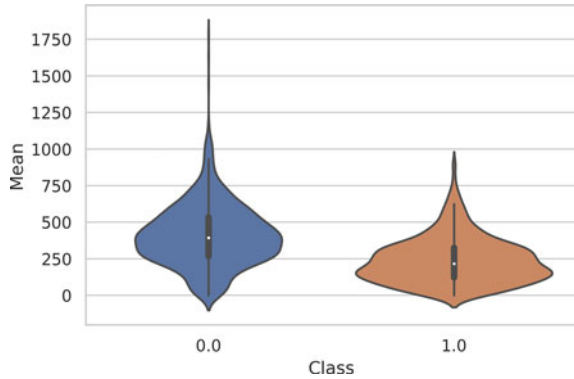
3 Proposed Methods

In this section we describe the proposed deep learning methodologies for identifying individuals with schizophrenia.

3.1 Bidirectional LSTM

Long short-term memory networks (LSTMs) [28] are a special type of recurrent unit that are capable of learning long-term dependencies. These networks inherently capture the relevant features from the input sequences.

A simple LSTM unit consists of an input gate (i_t), forget gate (f_t), and output gate (o_t) which regulate the amount of information (gradient) flow through cell state (c_t) at timestep t. The output of each LSTM cell is denoted by h_t, which is considered as the hidden state. The equations of the gates and cell state the output activation of each gate are shown below in Eq. 1. The cell state c_t behaves as a memory unit to LSTM, at each time step the previous cell state c_{t-1} is weighted with forget gate f_t to decide the amount of information to propagate forward and is combined with the input gate i_t and $\tilde{c}_t$ to form a new cell state.

$$
\begin{aligned}
\text{Forget gate: } a_\mathrm{f} &= W_\mathrm{f} \cdot [h_{t-1}, x_t] + b_\mathrm{f} & f_t &= \mathrm{sigmoid}(a_\mathrm{f}) \\
\text{Input gate: } a_\mathrm{i} &= W_\mathrm{i} \cdot [h_{t-1}, x_t] + b_\mathrm{i} & i_t &= \mathrm{sigmoid}(a_\mathrm{i}) \\
a_\mathrm{c} &= W_\mathrm{c} \cdot [h_{t-1}, x_t] + b_\mathrm{c} & \hat{c}_t &= \tanh(a_\mathrm{c}) \\
\text{Output gate: } a_\mathrm{o} &= W_\mathrm{o} \cdot [h_{t-1}, x_t] + b_\mathrm{o} & o_t &= \mathrm{sigmoid}(a_\mathrm{o}) \\
\text{Cell state: } c_t &= (f_t * c_{t-1}) + (i_t * \hat{c}_t) & \\
\text{Hidden state: } h_t &= o_t * \tanh(c_t) &
\end{aligned}
\tag{1}
$$

The associated weights and biases for each gate are denoted as W_f, W_i, W_c, W_o and b_f, b_i, b_c, b_o. These notations represent the weights and bias of forget, input, candidate cell state, output, and associated softmax layers respectively.

A Bidirectional LSTM network [29] is composed of two hidden layer consisting of LSTM units. One layer of the bidirectional network considers the input from one direction whereas the other layer considers the input in the opposite direction. The outputs of all the hidden states from both the forward and backward LSTM layers are concatenated together; i.e., let $\overrightarrow{h_t}$ be the hidden states calculated from the forward direction and $\overleftarrow{h_t}$ be the hidden states captured from the backward direction; then, the concatenated hidden vector is given by the following Eq. 2.

$$h_t = \text{concat}[\overrightarrow{h_t}, \overleftarrow{h_t}]. \tag{2}$$

Using bidirectional, the input is run in two directions which helps in preserving the information both from the beginning and ahead of the input signal. This helps in understanding the context better and extracts relevant features. The features extracted from the bidirectional LSTM are then passed on to an attention mechanism for further classification.

3.2 Attention Mechanism

In general LSTM models we pick the output of the final hidden state. This has a drawback of vanishing gradient because as length of input sequence increases the final hidden state need not necessarily contain information belonging to the beginning of the sequence. So, instead of focusing on the entire input sequence we focus on only those hidden units (encoded representation of input sequence from the Bidirectional LSTM layers) which are more sensitive toward predicting the output. If we pass the hidden state outputs to a regular dense layer the weights are static in nature, i.e., important tokens within input sequence may receive less scaling from the weights which are adjusted from previous input samples. To avoid this, we associate weights to the inputs showing the relevance of each input [30].

Instead of directly passing the outputs of the bidirectional layer to a fully connected layer we take a detour and pass the outputs of to a dense layer comprising of a single hidden unit which in turn is passed toward a softmax layer. The softmax layers calculate the probabilities of each encoded representation of the input. These probabilities are called *Attention Weights*. These weights are assigned to the encoded representation of the inputs; i.e., each token of the input signal is scaled by its relevance toward predicting the output. We then calculate the *Context Vector* which is the stated weighted average of the input representation. Therefore, the output of attention layer is the context vector which is calculated as below Eq. 3.
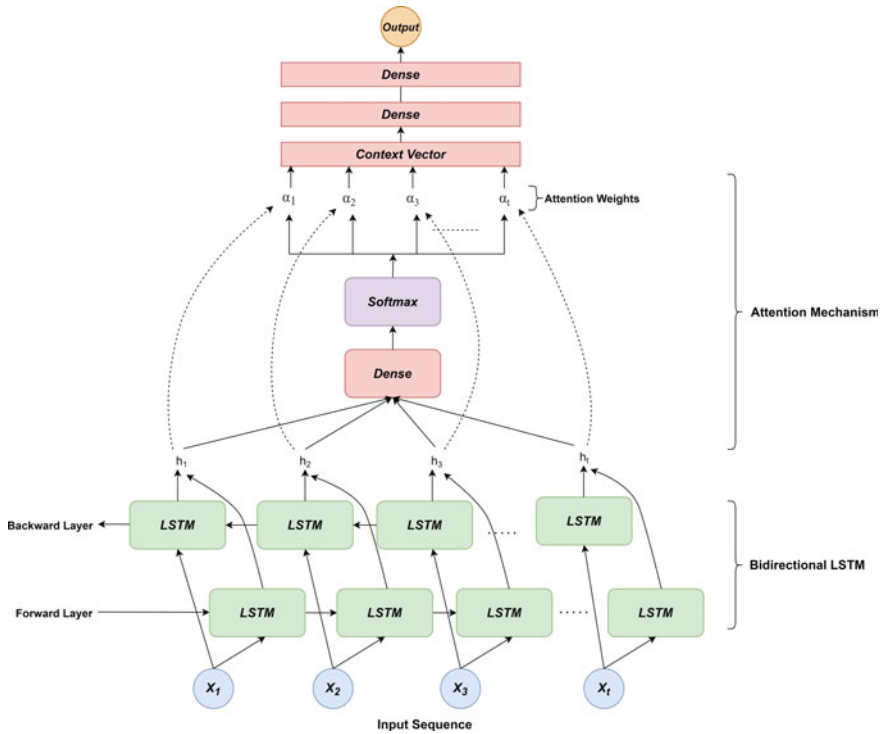
Fig. 2 Proposed architecture

$$\text{Context Vector} = \sum_{t'=1}^{d} \alpha(t') \cdot h(t') \tag{3}$$

where α is the attention weight vector which scales the hidden unit output h at time step t' and d represents the number of input features from the bidirectional layer. The complete proposed architecture for identifying patients suffering from schizophrenia is represented in Fig. 2.

The total number of LSTM units is set as 1024. Since we have employed bidirectional model, the total units become 2048 out of which, 1024 units correspond to the forward layer and the other correspond to the backward layer. The outputs of both the LSTM layers are concatenated together and result in hidden state h_i as shown in Fig. 2 where i corresponds to the intermediate hidden unit and h_t corresponds to the final input token. The resulting hidden units are then passed on the attention mechanism. The initial layer of the attention mechanism consists of a dense layer consisting of single unit with an activation of tanh. The output of this layer is then passed to a *softmax* activation layer. The softmax layer calculates the probabilities of each token of the input signal identifying the part of signal which plays more importance in deciding the outcome. These probabilities are referred to as *attention weights*

denoted by vector α. The resulting hidden states from the bidirectional layer are scaled against the *attention weights* weighing down the unimportant parts of the input signal and gives greater weightage to the tokens that play major role in determining the output and are represented as *context vector*. The output of the *context vector* is then passed on to a dense layer consisting of 1024 hidden units and then passed on to the output layer for classification. Further we interpreted the results by visualizing the attention weights and visually represent the signals and their corresponding important sub-parts of the sequence in the next section.

4 Experimental Results and Analysis

4.1 Results of the Proposed Bidirectional Architecture

The dataset [19] consists of actigraph data which essentially contains the intensity of acceleration due to activity at defined time periods. We split the data based on 3 h window intervals. These windows are extracted from general active time of an individual which is 9:00 a.m. to 21:00 p.m. This resulted in 4 samples per day for each subject. We developed a bidirectional LSTM network with attention mechanism which consists of LSTM layers which extract relevant features from the input signals and the attention mechanism scales up the important features which result in better classification of subjects suffering from schizophrenia to that of healthy subjects. The classification performance of the proposed method on schizophrenia dataset along with existing results is represented in Table 1.

Since, the data is imbalanced between diagnosed and healthy subjects, it is important to use different metrics. For our experiments, we have employed the standard accuracy, precision, recall, and $F1$-score as metrics. Additionally, we also report area under the curve for Receiver operating characteristic (ROC) curve, precision–recall (PR) curve and also Matthews Correlation Coefficient (MCC) which calculates the balanced measure in case of imbalanced datasets. The proposed model has shown a

Table 1 Classification performance results with comparison

Metrics (%)	Proposed method	LR [19]	RF [19]
Accuracy	86.60	–	–
Precision	87.12	**89.0**	83.6
Recall	**87.03**	–	83.7
$F1$-Score	**87.07**	–	83.6
AUC-ROC	**96.0**	81.0	–
AUC-PR	89.0	–	–
MCC	**72.59**	–	66.2
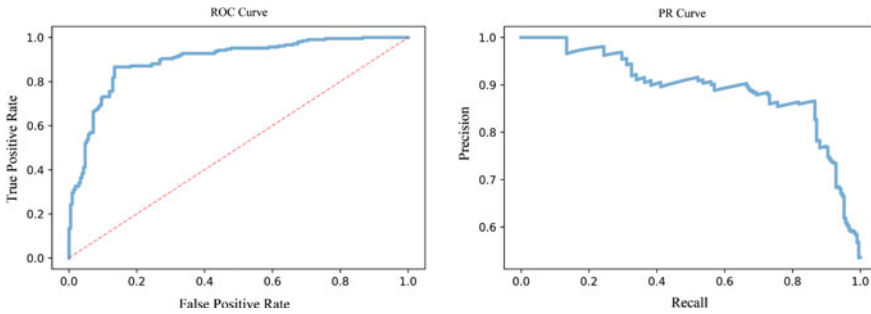
Fig. 3 The ROC and PR curve corresponding to the test data

test accuracy of about **86.60%** over 3 h windowed signals. The training and test data are split in the ratio of 80:20 respectively. The overall precision and recall on the test set is **87.12%** and **87.03%** respectively and resulted in an $F1$-score of **87.07%**. The area under the curve for ROC (AUC-ROC) and precision–recall curve is reported as **0.906** and **0.890** respectively which is represented in Fig. 3. Overall, the proposed method has resulted in decent scores over all the metrics although, the main goal is visualizing the features learned through the representation of the input signal and the important subsets of the signal which play the major role in identifying the underlying condition.

4.2 Visualization of Attention Weights for Learned Representation

For visualizing the relevant features learnt by the bidirectional layer of the input signal, we employ the attention weights. Weights are assigned to each actigraphy value within the input sequence stating the relevance of the value at that position. A higher weight to the intensity value indicates its importance in the identification of schizophrenic individuals. In Fig. 4 the left image represents the visualization of features extracted from 3 h activity cycle of randomly selected patients suffering with schizophrenia.

In Fig. 4, the forward directional LSTM has identified lower activity regions in the initial hour of the 3 h interval. The backward layer has identified a significant lower activity span between the second and third hour compared to other regions surrounding it. By emphasizing these relevant features from both the directions of the signal, the attention mechanism has assigned higher probabilities to the relevant ranges and lower probabilities to other regions. The attention weights are plotted as a heat map showing red color for higher probabilities and blue for lower probabilities. The relevant features are marked manually. In the left image which is of a patient having schizophrenia, there has been almost no activity between hours 1 and 3. The
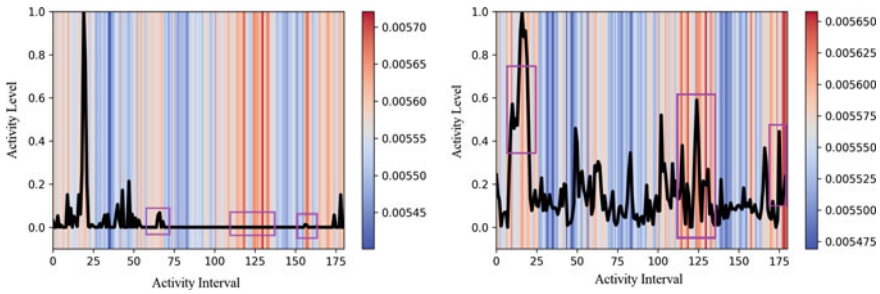
Fig. 4 Visualization of significant features extracted in determining the individuals suffering from schizophrenia and the healthy individuals

model has done a decent job in identifying the flat regions as highlighted and has predicted positive for schizophrenia.

In Fig. 4 the right image represents the visualization of features extracted from 3 h activity cycle of randomly selected healthy individuals. In the top image, higher activity values are highlighted manually in boxes. These regions are assigned higher probabilities as shown in the image. The model has identified the initial high activity regions pretty well and also identified in the intermediate ranges. Similarly in the right image of Fig. 4, the network has identified high activity regions and has classified healthy individuals accordingly based on the activity cycles. Therefore, though the performance metrics can only represent the decisive power of a model, it is important to understand why the model takes certain decisions even if the underlying model drifts over time with new training samples.

5   Conclusions and Future Scope

In this work, we propose a bidirectional LSTM layer with attention mechanism for the task of identifying individuals suffering from schizophrenia through visualization of distinguishing features separating the healthy and unhealthy. We have experimented on a publicly available dataset consisting of activity values measured over periods of time from healthy individuals as well as patients. We categorized the data based on 3 h window intervals sampled during general active times. Individuals suffering from schizophrenia have low motor activity thus resulting in lower activity values in the measured intervals whereas healthy individuals perform certain amount of activity compared to unhealthy individuals. The proposed model on such features has shown an overall accuracy of about 86.60%. We interpreted the learnt features of the model through visualization of attention weights which highlight the key regions within the input signal that have more weight in distinguishing lower and higher activity intervals and classify accordingly. Further, we seek to develop models based on the

combination of EEG and activity data that enable in the assistance and support for diagnosing mental health issues.

References

1. Zito MF, Marder SR (2020) Rethinking the risks and benefits of long-term maintenance in schizophrenia. Schizophr Res 225:77–81
2. Tandon R, Nasrallah HA, Keshavan MS (2009) Schizophrenia, "just the facts" 4. Clinical features and conceptualization. Schizophr Res 110(1–3):1–23
3. Keepers GA, Fochtmann LJ, Anzia JM, Benjamin S, Lyness JM, Mojtabai R, Servis M, Walaszek A, Buckley P, Lenzenweger MF et al (2020) The American psychiatric association practice guideline for the treatment of patients with schizophrenia. Am J Psychiatry 177(9):868–872
4. Garcia-Ceja E, Riegler M, Nordgreen T, Jakobsen P, Oedegaard KJ, Tørresen J (2018) Mental health monitoring with multimodal sensing and machine learning: a survey. Pervasive Mob Comput 51:1–26
5. Jakobsen P, Garcia-Ceja E, Riegler M, Stabell LA, Nordgreen T, Torresen J, Fasmer OB, Oedegaard KJ (2020) Applying machine learning in motor activity time series of depressed bipolar and unipolar patients compared to healthy controls. PLoS ONE 15(8):e0231995
6. Berle JO, Hauge ER, Oedegaard KJ, Holsten F, Fasmer OB (2010) Actigraphic registration of motor activity reveals a more structured behavioural pattern in schizophrenia than in major depression. BMC Res Notes 3(1):1–7
7. Winnebeck EC, Fischer D, Leise T, Roenneberg T (2018) Dynamics and ultradian structure of human sleep in real life. Curr Biol 28(1):49–59
8. Faedda GL, Ohashi K, Hernandez M, McGreenery CE, Grant MC, Baroni A, Polcari A, Teicher MH (2016) Actigraph measures discriminate pediatric bipolar disorder from attention-deficit/hyperactivity disorder and typically developing controls. J Child Psychol Psychiatry 57(6):706–716
9. Garcia-Ceja E, Riegler M, Jakobsen P, Torresen J, Nordgreen T, Oedegaard KJ, Fasmer OB (2018) Motor activity based classification of depression in unipolar and bipolar patients. In: 2018 IEEE 31st international symposium on computer-based medical systems (CBMS). IEEE, pp 316–321
10. Wee ZY, Yong SWL, Chew QH, Guan C, Lee TS, Sim K (2019) Actigraphy studies and clinical and biobehavioural correlates in schizophrenia: a systematic review. J Neural Transm 126(5):531–558
11. Krane-Gartiser K, Henriksen TE, Morken G, Vaaler AE, Fasmer OB (2018) Motor activity patterns in acute schizophrenia and other psychotic disorders can be differentiated from bipolar mania and unipolar depression. Psychiatry Res 270:418–425
12. Fasmer OB, Hauge E, Berle JØ, Dilsaver S, Oedegaard KJ (2016) Distribution of active and resting periods in the motor activity of patients with depression and schizophrenia. Psychiatry Investig 13(1):112
13. Meyer N, Faulkner SM, McCutcheon RA, Pillinger T, Dijk D-J, MacCabe JH (2020) Sleep and circadian rhythm disturbance in remitted schizophrenia and bipolar disorder: a systematic review and meta-analysis. Schizophr Bull 46(5):1126–1143
14. Dibner C, Schibler U, Albrecht U (2010) The mammalian circadian timing system: organization and coordination of central and peripheral clocks. Ann Rev Physiol 72:517–549
15. Henson P, Barnett I, Keshavan M, Torous J (2020) Towards clinically actionable digital phenotyping targets in schizophrenia. NPJ Schizophr 6(1):1–7
16. Delorme TC, Srivastava LK, Cermakian N (2020) Are circadian disturbances a core pathophysiological component of schizophrenia? J Biol Rhythms 35(4):325–339

17. Boeker M, Riegler MA, Hammer HL, Halvorsen P, Fasmer OB, Jakobsen P (2021) Diagnosing schizophrenia from activity records using hidden Markov model parameters. In: 2021 IEEE 34th international symposium on computer-based medical systems (CBMS). IEEE, pp 432–437
18. Reinertsen E, Osipov M, Liu C, Kane JM, Petrides G, Clifford GD (2017) Continuous assessment of schizophrenia using heart rate and accelerometer data. Physiol Meas 38(7):1456
19. Jakobsen P, Garcia-Ceja E, Stabell LA, Oedegaard KJ, Berle JO, Thambawita V, Hicks SA, Halvorsen P, Fasmer OB, Riegler MA (2020) Psykose: a motor activity database of patients with schizophrenia. In: 2020 IEEE 33rd international symposium on computer-based medical systems (CBMS). IEEE, pp 303–308
20. Afonso P, Figueira ML, Paiva T (2014) Sleep-wake patterns in schizophrenia patients compared to healthy controls. World J Biol Psychiatry 15(7):517–524
21. Bondugula RK, Udgata SK, Bommi NS (2021) A novel weighted consensus machine learning model for covid-19 infection classification using CT scan images. Arab J Sci Eng 1–12
22. Reinertsen E, Shashikumar SP, Shah AJ, Nemati S, Clifford GD (2018) Multiscale network dynamics between heart rate and locomotor activity are altered in schizophrenia. Physiol Meas 39(11):115001
23. Osipov M, Behzadi Y, Kane JM, Petrides G, Clifford GD (2015) Objective identification and analysis of physiological and behavioral signs of schizophrenia. J Ment Health 24(5):276–282
24. Dvey-Aharon Z, Fogelson N, Peled A, Intrator N (2015) Schizophrenia detection and classification by advanced analysis of EEG recordings using a single electrode approach. PLoS ONE 10(4):e0123033
25. Zhang L (2019) EEG signals classification using machine learning for the identification and diagnosis of schizophrenia. In: 2019 41st annual international conference of the IEEE engineering in medicine and biology society (EMBC). IEEE, pp 4521–4524
26. American Psychiatric Association (1994) Diagnostic and statistical manual of mental disorders, vol 4. American Psychiatric Association, Washington, DC
27. American Psychiatric Association (1987) Diagnostic and statistical manual of mental disorders DSM 3 R. American Psychiatric Press
28. Sundermeyer M, Schlüter R, Ney H (2012) LSTM neural networks for language modeling. In: Thirteenth annual conference of the international speech communication association
29. Huang Z, Xu W, Yu K (2015) Bidirectional LSTM-CRF models for sequence tagging. arXiv preprint arXiv:1508.01991
30. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. In: Advances in neural information processing systems, pp 5998–6008

Relevance of Frequent Pattern (FP)-Growth-Based Association Rules on Liver Diseases

Punyaban Patel, Borra Sivaiah, and Riyam Patel

Abstract Finding the most frequent itemsets and association rules is becoming a key focus of medical research as the demand for medical data processing grows. The association rule-based analysis assists researchers and analysts in determining the relationships and dependencies among distinct parameters in a liver dataset. This information can assist the patient in determining the source of their illness. There have been extensive researches conducted in order to find optimal association rule mining algorithm. As a result, this paper investigated and critically analyses an Apriori and FP-growth association rule discovery strategies and shows good accuracy. The objective of this work is to discover the strong association rules from liver disease dataset which increases the patient protective measures in order to preserve valuable people's lives.

Keywords Association rule mining · FP-growth · Liver diseases · Support · Confidence

1 Introduction

The progression of various complicated human diseases has been linked to irregularities in human metabolism. The leading cause of severe diseases are growing, and responsible for a large number of fatalities each year, showing to be a widespread problem for researchers, healthcare systems, individuals, and doctors together.

P. Patel (✉)
Department of Computer Science and Engineering, CMR Technical Campus, Kandlakoya, Hyderabad, India
e-mail: punyaban@gmail.com

B. Sivaiah
Department of Computer Science and Engineering, CMR College of Engineering and Technology, Kandlakoya, Hyderabad, India

R. Patel
Department of Computer Science and Engineering (AI and ML), SRM Institute of Science and Technology, SRM University, Kattankulathur, Chennai, India

Jaundice, hepatitis, hepatitis A, hepatitis B, hepatitis C, bile duct disorders, cirrhosis, tumour growth on liver and fatty liver disease are all examples of liver diseases. Over three million people die each year from liver disorders such as non-alcoholic steatohepatitis (NASH), non-alcoholic fatty liver disease (NAFLD), hepatocellular carcinoma (HCC), and liver cirrhosis, with 1.4 billion individuals presently overweight and life risk [1]. Currently no general treatment of NASH and NAFLD. Instead, individuals can lower their individual risk factors by making lifestyle modifications like exercising and dieting, or by using pharmacological remedies like antioxidants (like vitamin E), insulin sensitizers (like metformin), and cholesterol-lowering drugs (like statins) [2]. Despite their success, the fundamental mechanisms of action of most of these therapeutics, as well as the possible consequences of their use, remain poorly understood, particularly at the molecular and cellular levels.

Patients at risk of liver disease have traditionally been able to reduce their own risk by making long-term day-to-day routine changes such as walking, exercising as well as losing weight. A detailed study of weight gain and loss in insulin-resistant and sensitive people found that weight gain had health consequences that were either quickly increase or loss of weight [3].

Common metabolic illnesses are frequently complicated in nature, with significant multigene components and a plethora of underlying processes all leading to the same gross phenotype. Obesity, for example, has negative health consequences and is linked to some types of cancer [4], as well as being closely linked to NAFLD.

Systems medicine has shown to be consistent with previously published findings and may be utilised to perform in silico-simulations, generate hypotheses, and produce rational drugs. Workflow for computational biology that is iterative. Literature, experimentation, and publicly accessible databases are all used to collect data. The creation and refinement of computational models describing biological knowledge. The application of computer science approaches has been successfully used to examine relations in complicated disease on cellular levels, as detailed elsewhere [5].

In the field of computer science, the analysis of patient liver data is becoming extremely relevant. In a significant number of liver data sets, the liver data analysis is linked to the discovery of previously discovered hidden patterns (frequent itemset), associative structures.

Data mining is a discipline of computer science that identifies patterns in massive amounts of data and uses them to create forecasting model. The medical industry generates a significant volume of data that must be analysed. In the medical field, data mining improves the accuracy of healthcare trend prediction. Data mining techniques such as classification and prediction, clustering, association rule mining, and other ways may enhance medical data. WEKA includes several machine learning techniques for data mining tasks that leverage inbuilt algorithms to apply to lever illness data sets.

This paper is organised as where Sect. 1: Introduction, Sect. 2: Literature Review, Sect. 3: Association Rule Mining, Sect. 4: Matrices Used for Performance Measures, Sect. 5: Methodology, Sect. 6: Result and Analysis and Sect. 7 concluded the paper.

2 Literature Review

Many literatures are available in the library book, journals and web. The Apriori algorithm is widely used in medical research. These studies cover that how the association rule mining have been applied in Alzheimer's disease [6], cardiovascular disease [7, 8], liver cancer [9], lung cancer [10], oral cancer [11], type 2 diabetes [12, 13], infectious diseases (Ebola virus) [14]. In [15], there are some other diseases are enumerated for the study of which were used the Apriori algorithm or its modifications diseases like impotence, asthma, whooping cough, lupus, pregnancy, obesity, phenomenon Raynaud syndrome. The problems solved are also varied that is searching for unknown trends in disease, determining the nature of disease based on a prediction method, diagnosis of disease, predicting a patient's response to drug [16], early diagnosis and prevention of disease [17], prediction of illness's progress (course of disease), predicting the outcome of disease [7], identification of disease risk factors [13], identification of relationships between different medical operations, appointments, analyses and diagnoses [4, 5], extracting diagnostic patterns (sets of features, symptoms) and association rules in electronic medical database [18], and many others.

Diabetes may play an important role in the development of liver disease [19]. Patients with liver cancer are often diagnosed with diabetes, which accounts for 95% of all cases of diabetes. Even when age is taken into account, liver cancer and diabetes are diagnosed in the same individual more often than would be expected by chance. But no relationship has been shown between the two disorders. Diabetes-related factors such as steatosis, non-alcoholic fatty liver disease, and cirrhosis all raise the risk of liver cancer [20]. As a whole, evidence from observational studies shows that several diabetes medicines used to treat hyperglycemia may be connected to an increased risk of cancer in diabetics. In cancer patients, diabetes has also been related to a higher risk of mortality [20].

Chronic liver disease (CLD) is caused by many different of reasons, including alcohol, chronic hepatitis B, chronic hepatitis C, non-alcoholic steatohepatitis (NASH), malnutrition, toxins, and other tropical diseases, which are all frequent risk factors in central rural India [21]. According to [22], a decision tree analysis may be used to identify high-risk NAFLD patients. The primary properties of the tree structure may assist prevent non-alcoholic fatty liver disease (NAFLD) by identifying actionable community initiatives that can help improve NAFLD outcomes while reducing the burden on healthcare systems.

To model the links between distinct symptoms of observations and the diagnosis, the authors [23] used rough set theory. Different guidelines for diagnosis are formed by the attributes contained in the reduction obtained by a rough set method. The findings show that this data mining strategy for detecting kidney cancer based on historical data is realistic.

The authors [24] proposes the reduction of attributes eliminates overfitting of the problem and has a significant impact on the training period of neural networks.

Furthermore, using a neural network and a rough set hybrid technique to antici-
pate missing associations, eliminating conflicting objects enhances performance and
minimising training time.

In [25], the authors have proposed a model for patients with liver disease, the
Naive Bayes and KNN algorithms are intended to address predictive problems. The
result showed that the Naive Bayes algorithm outperforms the KNN algorithm by
incorporating six variables in the prediction model, resulting in a higher level of
accuracy.

For [26] accurate liver disease prediction, the modified convolutional neural
network-based liver disease prediction system (MCNN-LDPS) is introduced. Dimen-
sionality reduction is accomplished using modified PCA in the suggested study. The
score-based Artificial Fish Swarm Algorithm is used to identify the best features
(SAFSA). Information gain and entropy values are used as input variables in the
SAFSA method, which produced correct results based on a database of Indian liver
patients.

On the liver patient dataset, several classification techniques [27] such as logistic
regression, SMO, random forest algorithm, Naive Bayes, J48, and k-nearest neigh-
bour (KNN) are used to determine accuracy. With a software engineering model,
different classifier outcomes are compared using feature selection and without feature
selection techniques.

The authors [28] examined four different machine learning algorithms, including
random forest, logistic regression, and extra tree, to exclude irrelevant features from
the dataset using Pearson correlation coefficient-based features selection. According
to the simulation findings, the additional tree has the maximum accuracy.

3 Association Rule Mining

It emerged with a need to better understand market-basket data, which helped busi-
nesses to better understand purchasing behaviour and, as a consequent, target certain
segments of the market. The objective of Association Rule Mining (ARM) is to
find interesting rules from which new information may be extracted. It also allows
the user to locate, heuristically filter, and show implications or rules for subsequent
interpretation and decision making. According to the following association rule, the
mining process has been split into two phases:

Phase 1: Using a user-specified support threshold, identify the sets of frequently
occurring items, itemsets, or patterns within a set of transactions. Phase 2: Using
a user-specified confidence threshold, generate rules or inferences from the above
patterns found in Phase 1.

The first phase is the construction or mining of frequent itemsets, whereas Phase
2 is significantly more computationally demanding and involves the generation of
association rules. The computational complexity of this step is negotiable in contrast
to the first. There are two major concerns in the second phase. The first concern

is rule quantity, or an algorithm's capacity to create a large number of rules. The second problem is rule quality, which basically means that all of the rules are dull. The support and confidence measures are essential for eliminating unwanted itemsets and regulations from the mining process.

3.1 The Apriori Algorithm

In [29], it describes the first frequent itemset mining algorithm. The method was later enhanced and dubbed Apriori. The key advancement has been the development of the support of sets' monotonicity property. After the modification [30] it has been improved the monotonicity even more. The candidate generation strategy is used in the Apriori algorithm [31].

Algorithm
Apriori Algorithm

> Input: Items 'I', Large Items 'L', Support 'S', Confidence 'C'
> Output: Association Rules satisfying S and C 'R'
> Step 1: $R = \emptyset$;
> Step 2: (i) for each $I \in L$ do
>   (ii) for each $X \subset L$ such that $X \neq \Phi$ do
> $R = R \cup \{X \Rightarrow (I - X)\}$;
> Step 3: End

3.2 FP-Growth Algorithm

Without the necessity for candidate generation, Han et al. [32, 33] established an FP-growth approach for mining the full collection of frequent itemsets. It is employed as part of a divide-and-conquer strategy.

The database generates a list of often occurring objects in descending frequency order during the initial scan. Using a frequency descending order list, the database is compressed into a frequent pattern tree (FP-tree). The FP-tree is mined by first creating the conditional pattern basis for each of common length-1 pattern, then constructing its conditional FP-tree, and then mining iteratively in it.

To achieve growth of the pattern, the suffix pattern is concatenated with the common patterns created by a conditional FP-tree. To tackle the problem of finding long common patterns, the FP-growth algorithm recursively searched for shorter patterns before concatenating the suffix. By employing the least common elements

as a suffix, it improves selectivity. According to research conducted by FP-growth method significantly reduces search time.

Algorithm
FP-Growth Algorithm

Input: Dataset D, minsup, Itemset I, $J \subseteq I$
Output: $F(J)$
Step 1: $F(J) = \{\}$;
Step 2: for all $i \in I$ occurring in D
      {
         $F[J] = F[J]\{J \cup \{i\}\}$;
         $D_i = \{\}$;
         for all $j, I \in$ occurring in D such that $j > I$
         if $(\text{support}(J \cup \{i, j\}) \geq \text{minsupp})$
           $H = H \cup J$;
         for all $(\text{tid}, X) \in D$ with $I \in X$
           $D_i = D_i \cup (\text{tid}, X \cap H)$
           Compute $F[J \cup \{i\}]$;
           $F[J] = F[J] \cup F[J \cup \{i\}]$
      }
Step 3: End

4 Matrices Used for Performance Measures

The support confidence framework is used to capture a certain type of dependence among represented items in a database. This determines the uncertainty of an association rule with five factors, such as: confidence, support, leverage, lift, and conviction.

Support quantifies how frequently a rule applies to a given data set which can be used to filter out unnecessary rules, such as those that arise by coincidence or occur simply by chance. Confidence helps for measuring the reliability of the inference or interpretations that is made by a rule.

The lift value, which is the ratio of the rule's confidence to its expected confidence, is a measure of an association rule's importance. The product of the support values of the rule body and the rule head divided by the support of the rule body gives the expected confidence of a rule. It is defined as the percentage of database transactions that contain all elements in a given rule [29]. Lift is the ratio between the rule's confidence and the support of the itemset in the rule's consequent.

The support can be written as:

$$\text{Support}(X \rightarrow Y) = \text{Support}(Y \rightarrow X) = P(X \text{ and } Y) \tag{1}$$

where X, Y are the itemsets, $|XY|$ is the number transactions of itemset that contain both X and Y and $|D|$ represents the total number of transactions of itemset in the database.

The confidence [29] can be defined as

$$\text{Confidence}(X \rightarrow Y) = \frac{\text{Support}(X \rightarrow Y)}{\text{Support}(X)} \quad (2)$$

The leverage [29] can be defined as

$$\text{Leverage}(X \rightarrow Y) = P(X \text{ and } Y) - (P(X)P(Y)) \quad (3)$$

The conviction [29] can be defined as

$$\text{conviction}(X \rightarrow Y) = \frac{1 - \sup(Y)}{1 - \text{conf}(X \rightarrow Y)} \quad (4)$$

The lift [29] can be defined as

$$\text{lift}(X \rightarrow Y) = (Y \rightarrow X) = \frac{\text{conf}(Y \rightarrow X)}{\sup(X)} \quad (5)$$

For evaluating the association rules retrieved from the data, minimum support and confidence thresholds are chosen. If an itemset's support is greater than or equal to the minimal support value, it is considered common. One significant disadvantage of mining association rules is that it might be computationally expensive when apply in large data set.

5   Methodology

The data preprocessing is an essential step in knowledge discovery process. We applied data preprocessing to convert the continuous attributes of liver dataset into two nominal attributes (normal and abnormal) based on test range values of attributes of liver disease and non-liver disease patients. Both Apriori and FP-growth algorithms are applied to liver dataset and generated the association rules. The association rules are helpful for doctors to get which attributes are effecting the liver. The strong association rules are helpful for doctors to find the associations and correlation among the attributes in the liver dataset. The Apriori algorithm scans the liver data set for many times and takes more time for generating the association rules as compared to the FP-growth algorithm. FP-growth algorithm scans the liver dataset only two times. The process of association rules generation for liver diseases is shown in Fig. 1.
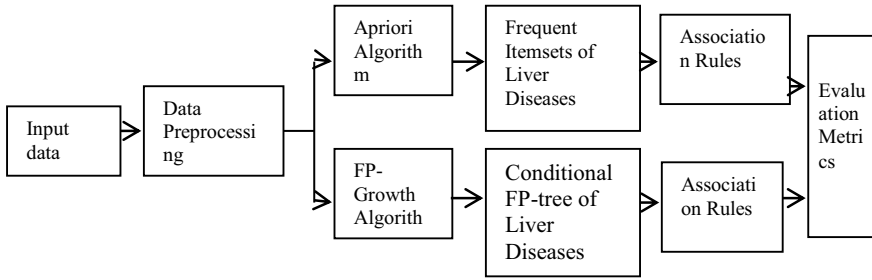
Fig. 1 Process of association rules generation for liver diseases

6   Results and Analysis

We used Kaggle's Indian liver patient data set (ILPD [34]). There are 441 male patient records and 142 female patient records in this collection. There are 416 liver patient records and 167 non-liver patient records in this data collection. The class attribute is used to categorise people into two groups (liver patient or not). The data set is shown in Table 2 and the properties of symptoms of patient's wise information plot is shown in Fig. 2.

Table 1 illustrate the best ten association rules generated by Apriori algorithm for the liver disease data set. Table 3 summarises the association rules generated by the FP-growth algorithm.

Using Apriori algorithm and FP-growth algorithm, the association rules has generated as shown in Tables 1 and 3, respectively. After analysis, the Apriori algorithm takes more number of cycles for generating frequent itemsets and association rules. But FP-growth algorithm takes only two cycles. So, we conclude that the FP-growth algorithm is better than the Apriori algorithm for the liver disease data set.
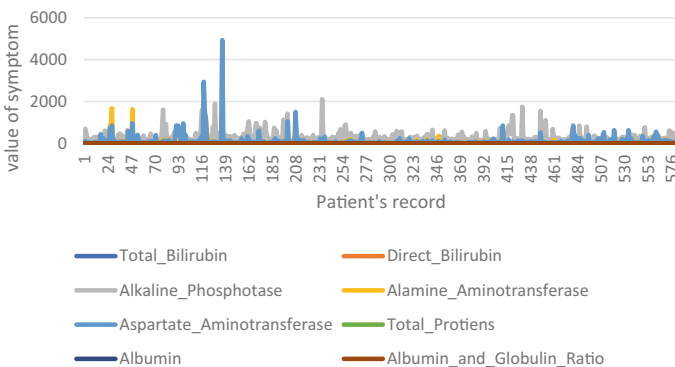
Fig. 2 No. of liver disease patient versus symptoms

Table 1 Top 10 association rules and their metrics for liver data set using Apriori algorithm

S. No.	Association rules	Confidence	Lift	Leverage	Conviction
1	[TP = normal]: 395 => Alkphos = abnormal 388	0.98	1	0	1.02
2	Sgpt = normal A_G_Rati = abnormal 391 => Alkphos = abnormal 384	0.98	1	0	1.01
3	A_G_Rati = abnormal 550 => Alkphos = abnormal 540	0.98	1	0	1.03
4	Gender = male A_G_Rati = abnormal 416 => Alkphos = abnormal 408	0.98	1	0	0.95
5	Sgpt = normal 421 => Alkphos = abnormal 412	0.98	1	0	0.87
6	Gender = male 441 => Alkphos = abnormal 431	0.98	1	0	0.83
7	A_G_Rati = abnormal selector_field = positive 395 => Alkphos = abnormal 386	0.98	1	0	0.81
8	DB = normal A_G_Rati = abnormal 389 => Alkphos = abnormal 380	0.98	1	0	0.8
9	DB = normal 418 => Alkphos = abnormal 408	0.98	1	0	0.78
10	selector_field = positive 416 => Alkphos = abnormal 406	0.98	1	0	0.78

Table 2 Liver data set description

S. No.	Attribute	Description
1	Age	Age of the patients
2	Gender	Sex of the patient
3	Total Bilirubin	Total Bilirubin in mg/dL
4	Direct Bilirubin	Conjugated Bilirubin in mg/dL
5	Alkaline_Phosphota	ALP in IU/L
6	Alamine_Aminotransferase	ALT in IU/L
7	Aspartate Aminotransferase	AST in IU/L
8	Total_Proteins	Total proteins g/dL
9	Albumin	Albumin in g/dL
10	Albumin_and_Globulin_Ratio	A/G ratio

7   Conclusion

With the uncontrolled increase in medical diagnostics and the use of automation for medical record analysis, researchers now have access to a massive amount of data. Nonetheless, data mining techniques are used by a variety of scholars to analyse

Table 3 Top 10 association rules of liver disease data set using FP-growth algorithm

S. No.	Association rules	Confidence	Lift	Leverage	Conviction
1	[Gender = male, DB = abnormal]: 142 => [TB = abnormal]: 142	1	2.1	0.13	74.29
2	[Sgot = abnormal, DB = abnormal]: 134 => [TB = abnormal]: 134	1	2.1	0.12	70.1
3	[DB = abnormal, Sgpt = abnormal]: 87 => [TB = abnormal]: 87	1	2.1	0.08	45.51
4	[Gender = male, Sgot = abnormal, DB = abnormal]: 117 => [TB = abnormal]: 117	1	2.1	0.1	61.21
5	[Gender = male, DB = abnormal, Sgpt = abnormal]: 76 => [TB = abnormal]	1	2.1	0.07	39.76
6	[Sgot = abnormal, DB = abnormal, Sgpt = abnormal]: 85 => [TB = abnormal]: 85	1	2.1	0.08	44.71
7	[Gender = male, Sgot = abnormal, DB = abnormal, Sgpt = abnormal]: 75 => [TB = abnormal]: 75	1	2.1	0.07	39.24
8	[DB = abnormal]: 165 => [TB = abnormal]: 164	0.99	2.08	0.15	43.16
9	[Gender = male, DB = abnormal, Sgpt = abnormal]: 76 => [Sgot = abnormal]: 75	0.99	2.15	0.07	20.6
10	[Gender = male, DB = abnormal, Sgpt = abnormal]: 76 => [TB = abnormal, Sgot = abnormal]: 75	0.99	3.06	0.09	25.75

and extract knowledge. This paper discover rules based on Apriori algorithm and FP-growth algorithm for detecting the effects of various parameters on liver dataset. It has also been found that association rule mining techniques have considerably advanced over the years. The FP-growth algorithm is better than Apriori algorithm in terms of number of scans. The investigation was successful in demonstrating the links between healthcare measures and the symptoms of liver disease. The work is designed to increase safety precautions based on the discovery and preservation of priceless human life.

References

1. Williams R (2019) Hepatology through the crystal ball. Hepatol Int 13:403–406
2. Singh S, Osna NA, Kharbanda KK (2017) Treatment options for alcoholic and non-alcoholic fatty liver disease: a review. World J Gastroenterol 23:6549–6570
3. Piening BD, Zhou W, Contrepois K et al (2018) Integrative personal omics profiles during periods of weight gain and loss. Cell Syst 6:157–170
4. Loos RJF, Bouchard C (2003) Obesity—is it a genetic disorder? J Intern Med 254:401–425
5. Mardinoglu A, Boren J, Smith U et al (2018) Systems biology in hepatology: approaches and applications. Nat Rev Gastroenterol Hepatol 15:365–377

6. Chavez R, Górriz JM, Ramirez J, Salas-Gonzalez D, Gbmez-Rio M (2011) Efficient mining of association rules for the early diagnosis of Alzheimer's disease. Phys Med Biol 56:6047–6063
7. Lakshmi KR, Krishna MV, Kumar SP (2013) Performance comparison of data mining techniques for predicting of heart disease survivability. Int J Sci Res Publ 3(6):1–10
8. Mirmozaffari M, Alinezhad A, Gilanpour A (2017) Data mining Apriori algorithm for heart disease prediction. Int J Comput Commun Instrum Eng (IJCCTE) 4(1):20–23
9. Pinheiro FMR (2013) Applying the Apriori and FP-growth association algorithms to liver cancer data. A thesis submitted in partial fulfillment of the requirement for the degree of Master of Science, University of Victoria, 1, p 72
10. Kanageswari S, Gladis D (2020) Generation of association rules of data mining for lung cancer by air pollution. Int J Eng Adv Technol (IJEAT) 9(3):2874–2880
11. Sharma N, Om H (2014) Early detection and prevention of oral cancer: association rules mining on investigation. WSEAS Trans Comput 13:1–8
12. Mao W, Mao J (2009) The application of Apriori-Gen algorithm in the association study in type 2 diabetes. In: Proceedings of 3rd international conference on bioinformatics and biomedical engineering (ICBBE), vol 4, no 1, pp 126–140
13. Ramezankhani A, Pournik O, Shahrabi J, Azizi F, Hadegh F (2015) An application of association rule mining to extract risk pattern for type 2 diabetes using Tehran lipid and glucose study database. Int J Endocrinol Metab 13(2):e25389
14. Go E, Lee S, Yoon T (2014) Analysis of ebolavirus with decision tree and Apriori algorithm. Int J Mach Learn Comput 4(6):543–548
15. Swathi P, Prajna B (2016) The effective procession of Apriori algorithm prescribed data mining on medical data. Int J Comput Sci Technol (IJCST) 7(3):22–26
16. Shang E, Duan J, Fan X, Tang Y, Ye L (2017) Association rule mining and statistic test over multiple datasets on TCM drug pairs. Int J Biomed Data Min 6(1):2–6
17. Shahebaz AhKh, Jabbar MA (2019) Improved classification techniques to predict the co-disease in diabetic mellitus patients using discretization and Apriori algorithm. Int J Innov Technol Explor Eng (IJITEE) 8(11):730–733
18. Karaolis M, Moutris JA, Papaconstantinou L, Pattichis CS (2009) Association rule analysis for the assessment of the risk of coronary heart events. In: Proceedings of the 31st annual international conference of the IEEE engineering in medicine and biology society, Minneapolis, MN, pp 6238–6241
19. Chang CK et al (2006) Occupational exposures and risks of liver cancer among Shanghai female textile workers—a case–cohort study. Int J Epidemiol 35:361–369
20. Giovannucci E et al (2010) Diabetes and cancer: a consensus report. CA Cancer J Clin 60:207–221
21. Banait S et al (2021) Risk factors for chronic liver disease in population of Central India: a case-control study from rural India. Egypt Liver J 11. Article number: 10
22. Yang T, Zhao B, Pei D (2021) Estimation of the prevalence of nonalcoholic fatty liver disease in an adult population in Northern China using the data mining approach. Dove Press 2021(14):3437–3445
23. Anitha A, Acharjya DP (2015) Neural network and rough set hybrid scheme for prediction of missing associations. Int J Bioinform Res Appl 11(6):503–524
24. Saleem Durai MA, Acharjya DP, Kannan A, Sriman Narayana Iyengar NC (2012) An intelligent knowledge mining model for kidney cancer using rough set theory. Int J Bioinform Res Appl 8(5–6):417–435
25. Hartatik MBT, Setyanto A (2020) Prediction for diagnosing liver disease in patients using KNN and Naïve Bayes algorithms. In: 2nd international conference on cybernetics and intelligent system (ICORIS). https://doi.org/10.1109/icoris50180.2020.9320797
26. Jeyalakshmi K, Rangaraj R (2021) Accurate liver disease prediction system using convolutional neural network. Indian J Sci Technol 14(17):1406–1421
27. Singh J, Bagga S, Kaur R (2020) Software-based prediction of liver disease with feature selection and classification techniques. Procedia Comput Sci 167:1970–1980. https://doi.org/10.1016/j.procs.2020.03.226

28. Rabbi MF et al (2020) Prediction of liver disorders using machine learning algorithms: a comparative study, pp 111–116. https://doi.org/10.1109/ICAICT51780.2020.9333528
29. Agarwal R, Imielinski T, Swami AN (1993) Mining association rules between sets of items in large databases. In: Proceedings of the 1993 ACM SIGMOD international conference on management of data
30. Borgelt C, Kruse R (2000) Induction of association rules: Apriori implementation. In: Proceedings of the fifteenth conference on computational statistics, Berlin, 24–28 Aug 2000, pp 395–400
31. Agarwal R, Mannila H, Srikant, R, Toivonen H, Verkamo A (1996) Fast discovery of association rules. In: Fayyad U, Piatetsky Shapiro G, Smyth P, Uthurusamy R (eds) Advances in knowledge discovery and data mining. MIT Press, pp 307–328
32. Han J, Pei J, Yin Y (2000) Mining frequent patterns without candidate generation. In: Proceedings of the 2000 ACM SIGMOD international conference on management of data, Dallas, TX, 16–18 May 2000, p 112
33. Haflidadottir S et al (2014) Long term follow-up and liver-related death rate in patients with non-alcoholic and alcoholic related fatty liver disease. BMC Gastroenterol 14:166
34. Pathan A et al (2018) Comparative study of different classification algorithms on ILPD dataset to predict liver disorder. Int J Res Appl Sci Eng Technol (IJRASET) 6(II)

Mist Assisted Cloud Computing Framework for Prediction of Hypertension Attack

Geetika Mudali, R. Priyadarshini, M. Kandpal, Rabindra K. Barik, and Diptendu S. Roy

Abstract Wearables and the Internet of Things (IoT) are becoming increasingly popular as a means of providing smart and affordable healthcare. In standard configurations, the cloud backend receives healthcare data and monitors and predicts illness diagnosis and well-being. Mist computing is a type of distributed computing that makes use of minimal power embedded processors as an intermediate layer between the fog and client layers. The mist layer is adhered tightly to the client layer's edge. Diagnostics for monitoring well-being and fitness might be transmitted from mist layer to fog layer. It was transmitted to the cloud server after processing in the fog layer. This paradigm results in a decrease in latency whilst increasing throughput. The purpose of this work is to create a mist-based deep learning model. The suggested model, *d-Mist*, takes data from individuals and uses mist computing framework to forecast their well-being statistics. This study discusses a significant aberration in health, notably hypertension attacks. Additionally, it conducted a thorough investigation of the suggested models' accuracy on standard datasets. The findings supported the suggested system and architecture's efficacy in accurately monitoring these essential health and fitness criterion.

Keywords Cloud computing · Fog computing · Mist computing · Hypertension attack · Data visualisation

G. Mudali
Department of CSE, National Institute of Science and Technology Berhampur, Brahmapur, India
e-mail: geetika@nist.edu

R. Priyadarshini
Department of CS and IT, C.V. Raman Global University, Bhubaneswar, India

M. Kandpal · R. K. Barik (✉)
KIIT Deemed to be University, Bhubaneswar, India
e-mail: rabindra.mnnit@gmail.com

D. S. Roy
Department of Computer Science and Engineering, National Institute of Technology, Meghalaya, Shillong, India
e-mail: diptendu.sr@nitm.ac.in

1 Introduction

The globe is increasing swiftly due to the Internet's computational and industrial expansion; the death rate has also grown. However, the number of lifestyle illnesses is rising. Type 2 diabetes, hypertension, heart attack and obesity are the few examples. Diet, stress, lack of physical activity, and environmental variables each have a role in various disorders. Symptoms of these disorders include irregular heartbeat, paralysis, cardiac arrest, chest discomfort and shortness of breath which necessitate immediate medical intervention [1, 2]. In 2014, over 410,000 Americans died from hypertension [3]. So quick monitoring and early diagnostic systems with low latency, fast reaction time, and high accuracy are required. The government is investing heavily in IT-based health solutions and services [1]. Because advanced computer facilities allow faster gathering, processing, and analysis of health data, and the transmission of extracted information to patients. However, these systems still have challenges with data transmission potential, managing and handling enormous amounts of big data, and functioning solution correctness. There are several cloud-based healthcare monitoring systems that store and analyse large amounts of data from many sources. It offers significant advantages such scalability, big data storage, cheap maintenance, implementation, and overall service costs [4]. Network-based bulk data transmission causes latency and affects system performance. The generation of vast volumes of data from IoT devices creates network latency and communication delay, especially in applications like health monitoring. Reduced communication and network delay caused by large data transfer across cloud servers is urgently required. The solution currently is a new computer paradigm called fog and mist computing [4–6].

Mist computing is a novel computing paradigm that helps cloud computing. This technique can help cloud computing handle massive amounts of data. With the aid of IoT, fog nodes manage effective services for storing and analysing huge data. Virtualization and data-centric resource management are the key technologies driving the entire process [4]. The mist aided cloud computing manages the workload in data centres across the network, namely, cloud. Unlike typical cloud computing, where user requests and applications flood the cloud server, mist computing regulates and optimises access to the cloud server for improved performance. It delivers extra duties like local storage, processing, and mining in the edge devices. So a smart gateway [7–9].

To summarise, mist layer may be utilised in healthcare applications to manage huge real-time data created by low-level sensors that continually monitor and evaluate data to forecast emotional attacks and detect disease. In light of this, the primary objectives of this proposed system is to present *d-Mist*, a mist-based healthcare telemonitoring system capable of remote diagnosis and prediction of hypertension attacks using characteristics extracted from the user's health-related data and observed symptoms.

This article is structured as follows. Section 2 comprises relevant work on the application of mist computing in the healthcare area for data processing, as well as the existing healthcare telemonitoring system. Section 3 is described with a full

overview of the framework. Section 4 contains the result and discussion. Section 5 summarises the paper's findings with concluding remarks.

2 Related Work

Cloud and fog computing are technologies that help novice/specialised users store, process, analyse, network, and transmit information. Like cloud computing data centres, fog nodes are closer to the clients or IoT layer for processing and providing many varieties of data [1]. Figure 1 shows the general cloud computing framework.

Numerous services are distributed in close proximity to end consumers/naive users. It is capable of determining the intermediary tier between fog and cloud servers. Local servers contain a similar amount of data [5, 6, 8, 10]. These are data packets containing intelligence that are delivered to cloud computers. These fog-assisted cloud computing frameworks can minimise both delay and latency by utilising these mechanisms [2, 4]. Although fog computing is superior to cloud computing in terms of performance for medical big data applications. It cannot alter fog in relation to cloud computing; rather, it must function concurrently as needed. Figure 2 illustrates the fog-assisted cloud computing architecture for IoT functions [5].
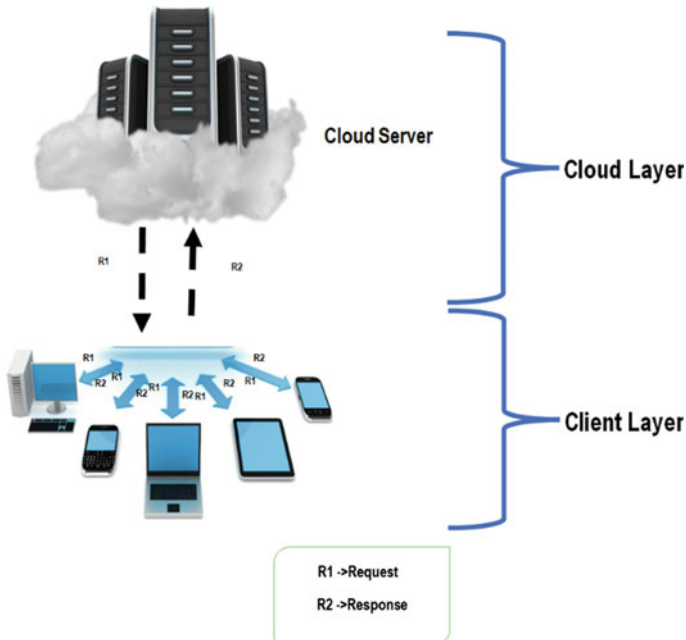
Fig. 1 General cloud computing framework for processing of data from client layer to cloud layer
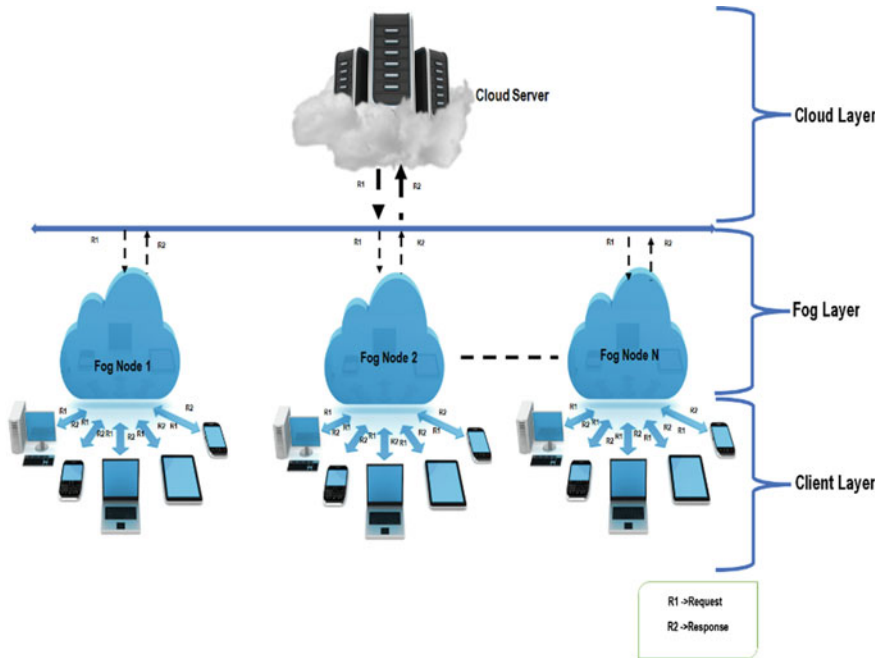
Fig. 2 General fog computing framework for processing of data from client layer to fog layer [5]

In a fog computing-based architecture, end users make queries to the fog layer, which processes them and returns the results to the customers/clients. There is a greater need for computing supplies elsewhere; they are supplied for analysis and storage at the cloud layer. When workloads are unbalanced across fog servers, all resource usage components, including fog servers, are well adjusted. Inefficient resource use and management, along with an unequal load across the servers, have lowered the system's QoS and increased energy consumption [3, 10–13].

Mist computing enables IoT layer computation at the network edge. To figure out the full information detected by the objects, microchips or microcontrollers are intimately coupled. They can then send the desired data to the system after figuring out the overall data. Mist aided cloud environment requires less transmission power and longer cloud storage. For example, the mist computing environment works near to the edge and physical devices, allowing for increased throughput and lower latency. So cloud storage has shrunk. As seen in Fig. 3, a reduction in necessary transmission power increases overall efficiency. Figure 3 describes the mist assisted cloud computing framework for data processing and storage [7].

They employed fog layer as a gateway for data mining, distributed storage, and alerting services at the network's edge. Raw signals are recorded, pre-processed, and wavelet transformed. To improve network capacity consumption and provide real-time notifications with reduced latency. A fog-based architecture for tele-healthcare monitoring of Parkinson disease patients was presented by Dubey et al. [14]. Data
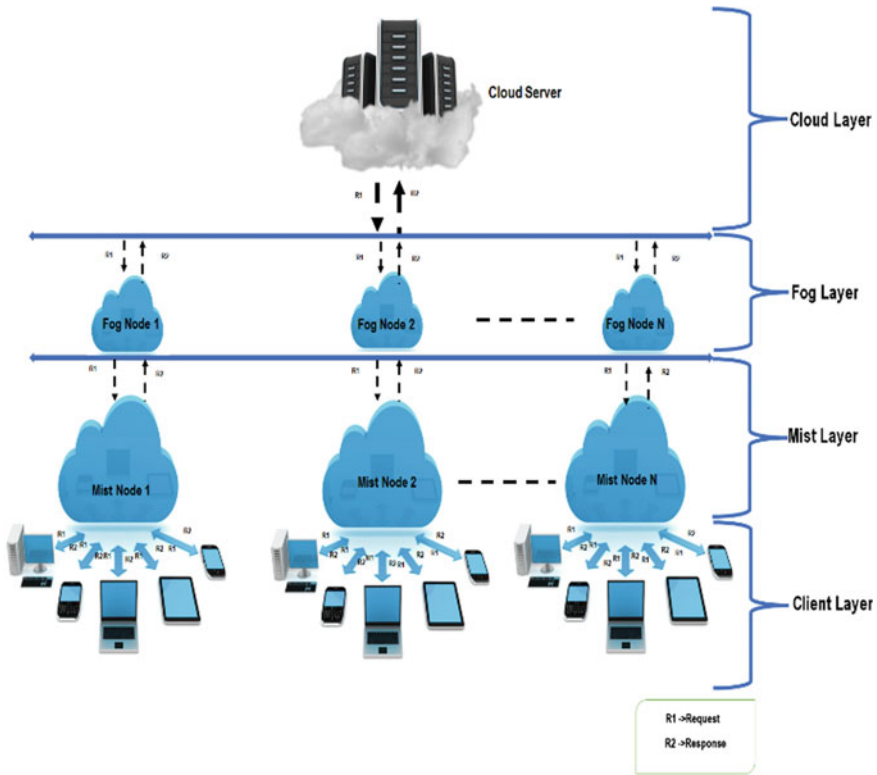
Fig. 3 General mist computing framework for processing of data from client layer to fog layer [7]

from a worn watch in its raw form. They were attempting to extract features from speech data such as loudness, short time energy, and zero-crossing rate, amongst others. In contrast to the previous author, they created their own Python code to extract data. Priyadarshini et al. [3] developed a novel architecture, i.e. *deepFog* which is monitoring system that alerts patients before an emergency occurs. The data are being compressed after it has been acquired in order to decrease transmission time. Lossy and lossless compression methods are used to compress the data. The data are analysed to make notification decisions. Smart phones with sensors are collected health data. They employ an intermediate fog layer between the end user and the cloud in order to increase the security of the data. Following a study of this literature survey, this paper is proposed the unique framework which is presented in Sect. 3.

3 Proposed Framework

The proposed framework, *d-Mist*, has four layers: client, Smart gateway mist, fog, and cloud. To diagnose hypertension, the client layer integrates all smart data gathering devices. Figure 4 depicts the method by which the collected data is transported to the intermediate mist layer for real-time processing, analysis, and prediction. The cloud layer receives the processed data, health-related outcomes, and diagnosis reports immediately. These data are then shared with topic experts, doctors, and family members who may take preventative steps and prompt action in emergency scenarios.

One case study is conducted in order to validate the model. The case study is conducted in order to learn more about the patient's behavioural characteristics. It is in charge of doing a risk assessment for a hypertension attack. All smart data
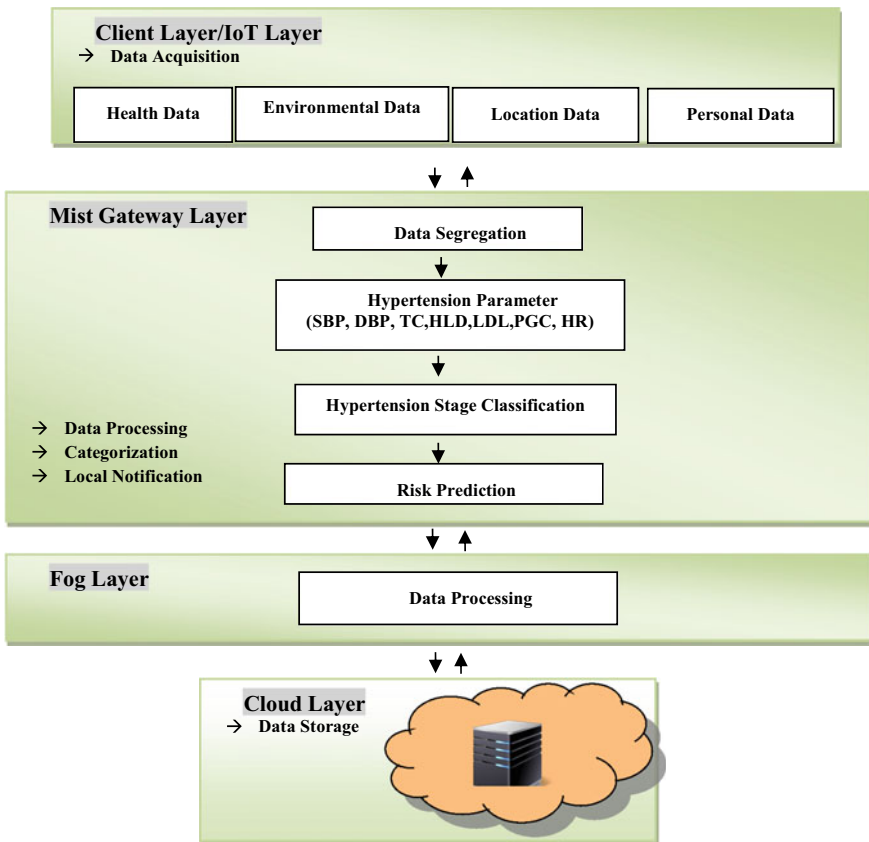
Fig. 4 Proposed d-Mist framework with a gateway layer amongst fog layer and client layer

collecting devices are integrated to collect health relevant feature data to identify hypertension attack.

4   Results and Discussion

Hypertension is a disorder that can cause cardiac failure, heart muscle thickening, and renal failure. Diastolic or systolic blood pressure readings of 90 mm Hg or 120 mm Hg for at least two measurements. However, abrupt changes in these numbers may produce hypertension stroke, leading to paralysis or coma. This disease's risk assessment is influenced by several variables, including environmental influences that might artificially raise blood pressure. There is evidence to suggest that clinical health information such as DBP, SBP, LDL and HDL have a stronger influence on hypertension than location and time do.

The three datasets are combined to create a data collection for forecasting hypertension attack risk. The hypertension attack training dataset is generated by merging chosen variables from three datasets: diabetes, stress, and kidney disease [15]. This dataset has 25 characteristics. Another property, hypertension attack, has a yes or no value, specifying whether or not the condition is present. Some characteristic attributes are picked from 24. A simple method is utilised to pick the feature. Using K-means clustering, the data are divided into two groups based on the hypertension field's yes or no value. Then the data clusters are evaluated to discover the contributing attribute distribution. If an attribute's data is discovered to be in the wrong cluster, it is not used in hypertension risk assessment training. The K-means algorithm levels data. 'Orange 3.4.4' is used for pre-analysis and data visualisation [16]. Orange is a Python-based open source data visualisation, machine learning, and data mining application. For exploratory data analytics and interactive data visualisation, use this tool. Basically, it is a Python module for data analytics. DBP and SBP distributions for two clusters are shown in Figs. 5 and 6. For this reason, these traits have a greater influence on created clusters. Figures 5 and 6 show the data values
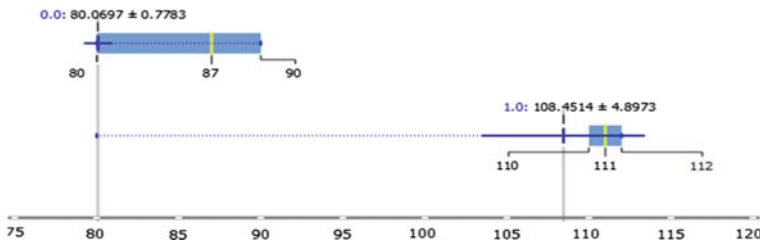
Fig. 5 A boxplot of DBP data shows the range and mean values after grouping the data using K-means clustering. The non-hypertension cluster is represented by the value 0.0, whereas the hypertension cluster is represented by the value 1.0. It is evident that the two sets of data do not have any similarities
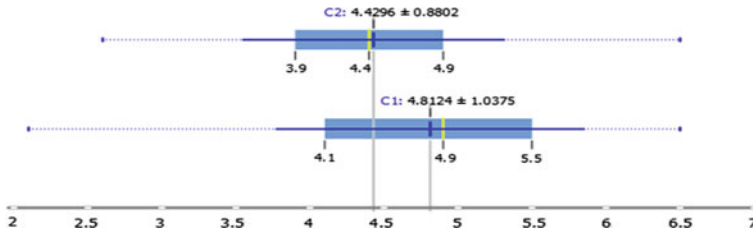
Fig. 6 Data distribution of RBC count in box plot, exhibiting range data, mean value after K-means clustering. The non-hypertension cluster is represented by the value 0.0, whereas the hypertension cluster is represented by the value 1.0. It has been demonstrated that the two sets of data are quite similar

shown for RBC and Potassium contents, respectively. These ideals clash. Positive and negative hypertension data are separated by these weak characteristics.

5 Conclusions

The globe has reached a great scientific and technical level. However, people's lifestyles are changing. Whilst physical activity has decreased, stress has increased. This is problematic since it accelerates the growth of lifestyle diseases like hypertension. It is currently a big worry of the government and doctors. Effective solutions may be created using modern computing and communication technologies. A possible mist aided cloud computing framework d-Mist is proposed here to identify a person's mental state, such as hypertension attack sufferers. It is developing a methodology to assess hypertension risk. Hypertension dataset is built and shown using Python-based machine learning and data mining toolkit Orange. This model will also be verified and validated using formal model.

References

1. Buyya R, Yeo CS, Venugopal S, Broberg J, Brandic I (2009) Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility. Future Gener Comput Syst 25(6):599–616
2. Barik RK, Dubey H, Mankodiya K (2017) SOA-FOG: secure service-oriented edge computing architecture for smart health big data analytics. In: 2017 IEEE global conference on signal and information processing (GlobalSIP), Nov 2017. IEEE, pp 477–481
3. Priyadarshini R, Barik RK, Dubey H (2018) DeepFog: fog computing-based deep neural architecture for prediction of stress types, diabetes and hypertension attacks. Computation 6(4):62
4. Barik RK, Tripathi A, Dubey H, Lenka RK, Pratik T, Sharma S et al (2018) MistGIS: optimizing geospatial data analysis using mist computing. In: Progress in computing, analytics and networking. Springer, Singapore, pp 733–742

5. Barik RK, Dubey H, Samaddar AB, Gupta RD, Ray PK (2016) FogGIS: fog computing for geospatial big data analytics. In: 2016 IEEE Uttar Pradesh section international conference on electrical, computer and electronics engineering (UPCON), Dec 2016. IEEE, pp 613–618

6. Barik RK, Dubey AC, Tripathi A, Pratik T, Sasane S, Lenka RK, Kumar V (2018) Mist data: leveraging mist computing for secure and scalable architecture for smart and connected health. Procedia Comput Sci 125:647–653

7. Barik RK, Misra C, Lenka RK, Dubey H, Mankodiya K (2019) Hybrid mist-cloud systems for large scale geospatial big data analytics and processing: opportunities and challenges. Arab J Geosci 12(2):32

8. Dutta A, Misra C, Barik RK, Mishra S (2021) Enhancing mist assisted cloud computing toward secure and scalable architecture for smart healthcare. In: Advances in communication and computational technology. Springer, Singapore, pp 1515–1526

9. Barik RK, Patra SS, Kumari P, Mohanty SN, Hamad AA (2021) A new energy aware task consolidation scheme for geospatial big data application in mist computing environment. In: 2021 8th international conference on computing for sustainable global development (INDIACom), Mar 2021. IEEE, pp 48–52

10. Tripathy SS, Barik RK, Roy DS (2022) Secure-M2FBalancer: a secure mist to fog computing-based distributed load balancing framework for smart city application. In: Advances in communication, devices and networking. Springer, Singapore, pp 277–285

11. Barik RK, Dubey H, Mankodiya K, Sasane SA, Misra C (2019) GeoFog4Health: a fog-based SDI framework for geospatial health big data analysis. J Ambient Intell Humaniz Comput 10(2):551–567

12. Barik RK, Priyadarshini R, Dubey H, Kumar V, Yadav S (2018) Leveraging machine learning in mist computing telemonitoring system for diabetes prediction. In: Advances in data and information sciences. Springer, Singapore, pp 95–104

13. Tripathy SS, Roy DS, Barik RK (2021) M2FBalancer: a mist-assisted fog computing-based load balancing strategy for smart cities. J Ambient Intell Smart Environ 1–15 (Preprint)

14. Dubey H, Goldberg JC, Abtahi M, Mahler L, Mankodiya K (2015) EchoWear: smartwatch technology for voice and speech treatments of patients with Parkinson's disease. In: Proceedings of the conference on wireless health, Oct 2015, pp 1–8

15. https://archive.ics.uci.edu/ml/datasets/Chronic_Kidney_Disease

16. Demšar J, Curk T, Erjavec A, Gorup Č, Hočevar T, Milutinovič M, Možina M, Polajnar M, Toplak M, Starič A, Štajdohar M (2013) Orange: data mining toolbox in Python. J Mach Learn Res 14(1):2349–2353

Author Index